

**- EPE -
NEW NAME
NEW DESIGN!**



Practical Electronics

The UK's premier electronics and computing maker magazine

Audio Out
LS3/5a
crossover

**Electronic
Building Blocks**
Digital mains meter

Circuit Surgery
Understanding
Logic levels

Micromite
Serial data
communication



Awesome Audio DSP



Tiny PIC circuits

WIN!
Microchip
MPLAB PICKit 4
In-Circuit
Debugger

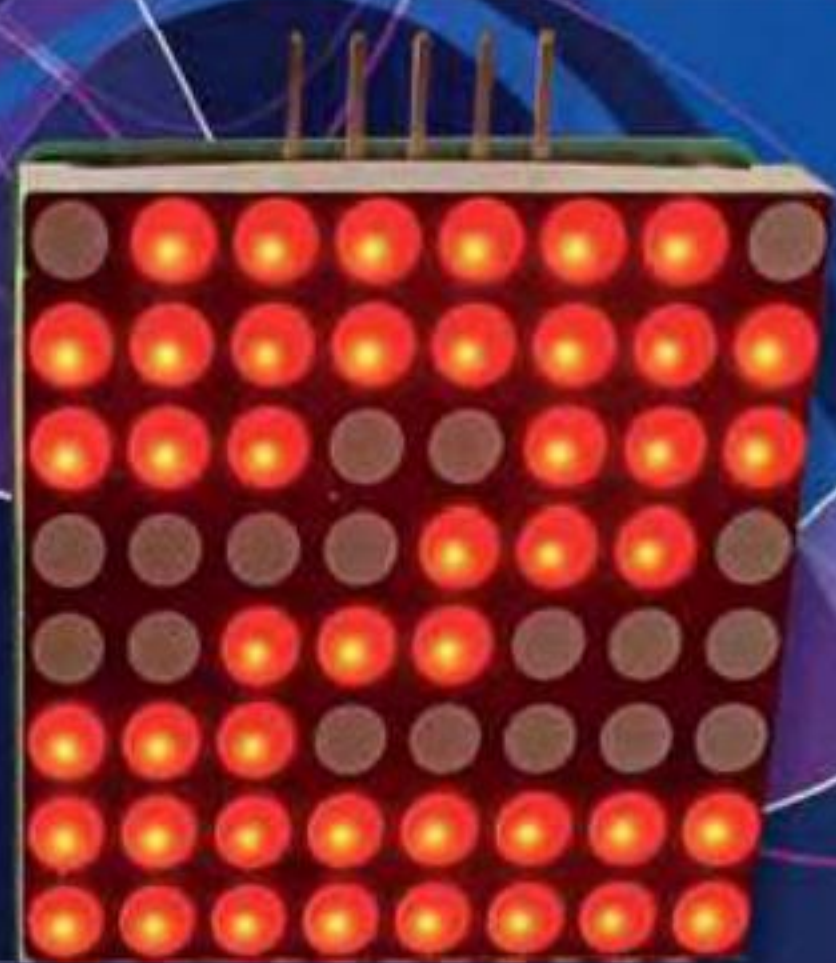
Bipolar stepper motor drivers



WIN!
\$50 of PCB prototyping
from PCBWay



Isolated Serial Link



Controlling an 8x8 LED matrix

PLUS!

PIC n' Mix – Temperature and humidity sensing
Net Work – The growth of smart metering
Techno Talk – Energy from the heavens: at night!



Using your Maximite



Jan 2020 £4.99



www.epemag.com



@practicalelec



practicalelectronics

With Over 250 Million Units in Stock, We've Got You Covered

Delivering You the World's Largest Inventory of Microchip Products



www.microchip.com/learn-more



The Microchip name and logo and the Microchip logo are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.
All other trademarks are the property of their registered owners.
© 2018 Microchip Technology Inc. All rights reserved. DS00002552A. MEC2226Eng10/18

Projects and Circuits

Audio DSP – Part 1 by Phil Prosser (design) and Nicholas Vinen (words) **14**
This is an exciting project, but what exactly is it? DSP? active crossover? – or perhaps an 8-channel parametric equaliser? In fact, it is all of these... and more.

Isolated Serial Link by Tim Blythman **24**
Want to communicate with a micro connected to mains or a high-voltage supply? Risky! Not just to the device, but to you as well! Here's the safe way to do it.

Four-channel High-current DC Fan and Pump Controller – Part 2 **32**
by Nicholas Vinen
PCB assembly, wiring and adjusting settings to suit your installation.

Colour Maximite Computer – Part 3 by Phil Boyce **40**
Introduction to using the Colour Maximite Computer with MMBASIC.

Series, Features and Columns

The Fox Report by Barry Fox **8**
Peripheral vision at the movies

Techno Talk by Mark Nelson **10**
Fresh nonsense

Net Work by Alan Winstanley **12**
A reminder of the most important law of computing. Back up now, because there are only two kinds of computer – ones that have failed, and ones that are going to fail.

PIC n' MIX by Mike Hibbett **45**
Small, cheap and powerful – Part 5

Audio Out by Jake Rothman **49**
PE Mini Monitor Crossover – Part 1

Using Stepper Motors by Paul Cooper **54**
Bipolar stepper motor drivers

Max's Cool Beans by Max The Magnificent **59**
Don't just think about it, make it!

Circuit Surgery by Ian Bell **62**
Logic levels – Part 1

Make it with Micromite by Phil Boyce **66**
Part 12: Writing serial data to external modules

Electronic Building Blocks by Julian Edgar **76**
Digital mains voltmeter and ammeter display

Regulars and Services

Subscribe to Practical Electronics and save money **4**

Wireless for the Warrior **6**

Reader services – Editorial and Advertising Departments **7**

Editorial **7**
Has it really been a year?

Exclusive Microchip reader offer **11**
Win a Microchip MPLAB PICKit 4 In-Circuit Debugger

Practical Electronics back issues CD-ROM – great 15-year deal! **17**

Practical Electronics CD-ROMS for electronics **70**
A superb range of CD-ROMs for hobbyists, students and engineers

Direct Book Service **73**
Build your library of carefully chosen technical books

PE Teach-In 8 **77**

Practical Electronics PCB Service **78**
PCBs for Practical Electronics projects

Classified ads and Advertiser index **79**

Next month! – highlights of our next issue of Practical Electronics **80**

Stepper motor image on cover and contents page courtesy of Pololu Robotics & Electronics, pololu.com

© Electron Publishing Limited 2019

Copyright in all drawings, photographs, articles, technical designs, software and intellectual property published in *Practical Electronics* is fully protected, and reproduction or imitation in whole or in part are expressly forbidden.

The February 2020 issue of *Practical Electronics* will be published on Thursday, 2 January 2020 – see page 80.



Quasar Electronics Limited
PO Box 6935, Bishops Cleeve
CM23 4WP, United Kingdom

Tel: 01279 467799
E-mail: sales@quasarelectronics.co.uk
Web: quasarelectronics.co.uk

All prices INCLUDE 20.0% VAT. Free UK delivery on orders over £48
Postage & Packing Options (Up to 0.5Kg gross weight): UK Standard 3-7
Day Delivery - £4.95; UK Mainland Next Day Delivery - £9.95; Europe
(EU) - £12.95; Rest of World - £14.95 (up to 0.5Kg).
!! Order online for reduced price postage and fast despatch !!
Payment: We accept all major credit/debit cards. Make cheques/PO's
payable to Quasar Electronics Limited.
Please visit our online shop now for full details of over 1000 electronic
kits, projects, modules and publications. Discounts for bulk quantities.



QuasarElectronics.co.uk

Solutions for Home, Education & Industry Since 1993



Official Main Dealer stocking the full range of Kits, Modules, Robots, Instruments, tools and much, much more...

PIC & ATMEL Programmable

We have a wide range of PIC, ATMEL Arduino and Raspberry Pi projects.

PIC Programmer & Experimenter Board

Great learning tool. Includes programming examples and a reprogrammable 16F627 Flash Microcontroller. Test buttons & LED indicators. Software to compile & program your source code is included. Supply: 12-15Vdc. Pre-assembled and ready to use.
Order Code: VM111 - ~~£38.88~~ **£35.94**



USB PIC Programmer and Tutor Board

The only tutorial project board you need to take your first steps into Microchip PIC programming using a PIC16F882 (included). Later you can use it for more advanced programming. Programs all the devices a Microchip PICKIT2™ can! Use the free Microchip tools for PICKIT2™ & MPLAB® IDE environment.
Order Code: EDU10 - **£46.74**



USB /Serial Port PIC Programmer

Fast programming. Wide range of PICs supported (see website for details). Free Windows software & ICSP header cable. USB or Serial connection. ZIF Socket, leads, PSU not included.
Kit Order Code: 3149EKT - ~~£49.96~~ **£29.95**
Assembled Order Code: AS3149E - **£44.95**
Assembled with ZIF socket Order Code: AS3149EZIF - ~~£74.96~~ **£49.95**



PICKIT™2 USB PIC Programmer Module

Versatile, low cost, PICKIT™2 Development Programmer. Programs all the devices a Microchip PICKIT2 programmer can. Onboard sockets & ICSP header. USB powered.
Assembled Order Code: VM203 - **£35.94**



Controllers & Loggers

Here are just a few of the controller and data acquisition and control units we have. See website for full details. 12Vdc PSU for all units: Order Code 660.446UK £10.68

USB Experiment Interface Board

Updated Version! 5 digital inputs, 8 digital outputs plus two analogue inputs and two analogue outputs. 8 bit resolution. DLL.
Kit Order Code: K8055N - ~~£39.95~~ **£22.20**
Assembled Order Code: VM110N - **£35.94**



2-Channel High Current UHF RC Set

State-of-the-art high security. Momentary or latching relay outputs rated to switch up to 240Vac @ 12 Amps. Range up to 40m. 15 Tx's can be learnt by one Rx. Kit includes one Tx (more available separately). 9-15Vdc.
Kit Order Code: 8157KT - **£44.95**
Assembled Order Code: AS8157 - **£49.96**



Computer Temperature Data Logger

Serial port 4-ch temperature logger. °C/°F. Continuously log up to 4 sensors located 200m+ from board. Choice of free software applications downloads for storing/using data. PCB just 45x45mm. Powered by PC. Includes one DS18S20 sensor.
Kit Order Code: 3145KT - ~~£19.95~~ **£16.97**
Assembled Order Code: AS3145 - **£19.96**
Additional DS18S20 Sensors - **£4.96 each**



8-Channel Ethernet Relay Card Module

Connect to your router with standard network cable. Operate the 8 relays or check the status of input from anywhere in world. Use almost any internet browser, even mobile devices. Email status reports, programmable timers... Test software & DLL online.
Assembled Order Code: VM201 - **£130.80**



Computer Controlled / Standalone Unipolar Stepper Motor Driver

Drives any 5-35Vdc 5, 6 or 8-lead unipolar stepper motor rated up to 6 Amps. Provides speed and direction control. Operates in stand-alone or PC-controlled mode for CNC use. Connect up to six boards to a single parallel port. Board supply: 9Vdc. PCB: 80x50mm.
Kit Order Code: 3179KT - **£15.26**
Assembled Order Code: AS3179 - **£22.26**



Bidirectional DC Motor Speed Controller

Control the speed of most common DC motors (rated up to 32Vdc/5A) in both the forward and reverse directions. The range of control is from fully OFF to fully ON in both directions. The direction and speed are controlled using a single potentiometer. Screw terminal block for connections. PCB: 90x42mm.
Kit Order Code: 3166KT - **£19.99**
Assembled Order Code: AS3166 - **£29.99**



8-Ch Serial Port Isolated I/O Relay Module

Computer controlled 8 channel relay board. 5A mains rated relay outputs and 4 opto-isolated digital inputs (for monitoring switch states, etc). Useful in a variety of control and sensing applications. Programmed via serial port (use our free Windows interface, terminal emulator or batch files). Serial cable can be up to 35m long. Includes plastic case 130x100x30mm. Power: 12Vdc, 500mA.
Kit Order Code: 3108KT - **£74.95**
Assembled Order Code: AS3108 - **£89.95**



8-Channel RF Remote Control Set

Control 8 onboard relays with included RF remote control unit. Toggle or momentary mode for each output. Up to 50m range. Board Supply: 12Vac, 500mA
Assembled Order Code: VM118 - **£71.94**



Temperature Monitor & Relay Controller

Computer serial port temperature monitor & relay controller. Accepts up to four Dallas DS18S20 / DS18B20 digital thermometer sensors (1 included). Four relay outputs are independent of the sensors giving flexibility to setup the linkage any way you choose. Commands for reading temperature / controlling relays are simple text strings sent using a simple terminal or coms program (e.g. HyperTerminal) or our free Windows application. Supply: 12Vdc.
Kit Order Code: 3190KT - ~~£79.96~~ **£47.95**
Assembled Order Code: AS3190 - **£59.95**



3x5Amp RGB LED Controller with RS232

3 independent high power channels. Preprogrammed or user-editable light sequences. Standalone or 2-wire serial interface for microcontroller or PC communication with simple command set. Suits common anode RGB LED strips, LEDs, incandescent bulbs. 12A total max. Supply: 12Vdc. 69x56x18mm
Kit Order Code: 8191KT - **£24.95**
Assembled Order Code: AS8191 - **£27.95**



**Card Sales
& Enquiries**

**01279
467799**

Build your dream modular analogue synth from our range of over 80 kits, including front panels, components and PCBs

Components for the synth builder and hobbyist

- Resistors and Capacitors
- Diodes, Transistors and ICs
- Switches and Potentiometers
- Knobs, sockets, connectors, leads and more

Order online at
www.soundtronics.co.uk
 Unit 29, Webb Ellis Business Park
 Rugby, Warwickshire
 CV21 2NP
 Tel: 01788 878 002



Soundtronics MIDI Ultimate Analogue Synth

As featured in Practical Electronics
 February - July 2019. Available as PCB, components, case or full kits

- Three VCOs
- Mixer
- 12dB VCF
- Two LFOs
- VCA
- Two ADSRs
- Sample & Hold
- Noise Source
- Repeat Gate Generator
- Echo Effect
- MIDI Interface
- Headphone Amplifier
- Toggle Switch Patching



Fluke/Philips PM3092 Oscilloscope
 2+2 Channel 200MHz Delay TB,
 Autoset etc – £250

STEWART OF READING

17A King Street, Mortimer, near Reading, RG7 3RS
 Telephone: **0118 933 1111** Fax: **0118 933 2375**

USED ELECTRONIC TEST EQUIPMENT

Check website **www.stewart-of-reading.co.uk**

(ALL PRICES PLUS CARRIAGE & VAT)

Please check availability before ordering or calling in



HP 54600B Oscilloscope
 Analogue/Digital Dual Trace 100MHz
 Only £75, with accessories £125

LAMBDA GENESYS	PSU GEN100-15 100V 15A Boxed As New	£400	Marconi 2305	Modulation Meter	£250
LAMBDA GENESYS	PSU GEN50-30 50V 30A	£400	Marconi 2440	Counter 20GHz	£295
IFR 2025	Signal Generator 9kHz – 2.51GHz Opt 04/11	£900	Marconi 2945/A/B	Communications Test Set Various Options	POA
IFR 2948B	Communication Service Monitor Opts 03/25 Avionics	POA	Marconi 2955	Radio Communications Test Set	£595
IFR 6843	Microwave Systems Analyser 10MHz – 20GHz	POA	Marconi 2955A	Radio Communications Test Set	£725
R&S APN62	Syn Function Generator 1Hz – 260kHz	£295	Marconi 2955B	Radio Communications Test Set	£800
Agilent 8712ET	RF Network Analyser 300kHz – 1300MHz	POA	Marconi 6200	Microwave Test Set	£1,500
HP8903A/B	Audio Analyser	£750 – £950	Marconi 6200A	Microwave Test Set 10MHz – 20GHz	£1,950
HP8757D	Scaler Network Analyser	POA	Marconi 6200B	Microwave Test Set	£2,300
HP3325A	Synthesised Function Generator	£195	Marconi 6960B	Power Meter with 6910 sensor	£295
HP3561A	Dynamic Signal Analyser	£650	Tektronix TDS3052B	Oscilloscope 500MHz 2.5GS/s	£1,250
HP6032A	PSU 0-60V 0-50A 1000W	£750	Tektronix TDS3032	Oscilloscope 300MHz 2.5GS/s	£995
HP6622A	PSU 0-20V 4A Twice or 0-50V 2A Twice	£350	Tektronix TDS3012	Oscilloscope 2 Channel 100MHz 1.25GS/s	£450
HP6624A	PSU 4 Outputs	£400	Tektronix 2430A	Oscilloscope Dual Trace 150MHz 100MS/s	£350
HP6632B	PSU 0-20V 0-5A	£195	Tektronix 2465B	Oscilloscope 4 Channel 400MHz	£600
HP6644A	PSU 0-60V 3.5A	£400	Farnell AP60/50	PSU 0-60V 0-50A 1kW Switch Mode	£300
HP6654A	PSU 0-60V 0-9A	£500	Farnell XA35/2T	PSU 0-35V 0-2A Twice Digital	£75
HP8341A	Synthesised Sweep Generator 10MHz – 20GHz	£2,000	Farnell AP100-90	Power Supply 100V 90A	£900
HP83630A	Synthesised Sweeper 10MHz – 26.5 GHz	POA	Farnell LF1	Sine/Sq Oscillator 10Hz – 1MHz	£45
HP83624A	Synthesised Sweeper 2 – 20GHz	POA	Racal 1991	Counter/Timer 160MHz 9 Digit	£150
HP8484A	Power Sensor 0.01-18GHz 3nW-10µW	£75	Racal 2101	Counter 20GHz LED	£295
HP8560E	Spectrum Analyser Synthesised 30Hz – 2.9GHz	£1,750	Racal 9300	True RMS Millivoltmeter 5Hz – 20MHz etc	£45
HP8563A	Spectrum Analyser Synthesised 9kHz – 22GHz	£2,250	Racal 9300B	As 9300	£75
HP8566B	Spectrum Analyser 100Hz – 22GHz	£1,200	Solartron 7150/PLUS	6½ Digit DMM True RMS IEEE	£65/£75
HP8662A	RF Generator 10kHz – 1280MHz	£750	Solartron 1253	Gain Phase Analyser 1mHz – 20kHz	£600
Marconi 2022E	Synthesised AM/FM Signal Generator 10kHz – 1.01GHz	£325	Solartron SI 1255	HF Frequency Response Analyser	POA
Marconi 2024	Synthesised Signal Generator 9kHz – 2.4GHz	£800	Tasakago TM035-2	PSU 0-35V 0-2A 2 Meters	£30
Marconi 2030	Synthesised Signal Generator 10kHz – 1.35GHz	£750	Thurlby PL320QMD	PSU 0-30V 0-2A Twice	£160 – £200
Marconi 2023A	Signal Generator 9kHz – 1.2GHz	£700	Thurlby TG210	Function Generator 0.002-2MHz TTL etc Kenwood Badged	£65



HP/Agilent HP 34401A Digital
 Multimeter 6½ Digit £325 – £375

HP33120A
 HP53131A
 HP53131A
 Audio Precision
 Datron 4708
 Druck DPI 515
 Datron 1081
 ENI 325LA
 Keithley 228
 Time 9818

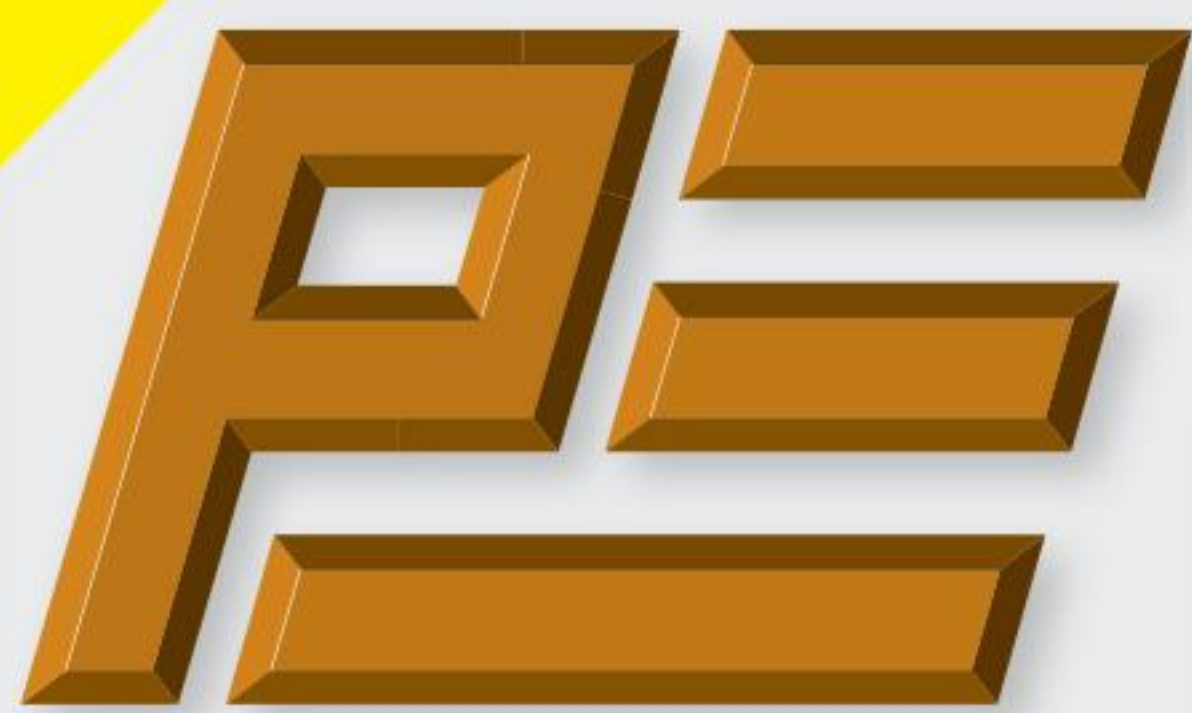
Function Generator 100 microHz – 15MHz
 Universal Counter 3GHz Boxed unused
 Universal Counter 225MHz
 SYS2712 Audio Analyser – in original box
 Autocal Multifunction Standard
 Pressure Calibrator/Controller
 Autocal Standards Multimeter
 RF Power Amplifier 250kHz – 150MHz 25W 50dB
 Voltage/Current Source
 DC Current & Voltage Calibrator

£350
 £600
 £350
 POA
 POA
 £400
 POA
 POA
 POA
 POA



Marconi 2955B Radio
 Communications Test Set – £800

- EPE -
NEW NAME
NEW DESIGN!



Practical Electronics

UK readers **SAVE £1** on every issue



Take out a one-year subscription and save **£10** over the year.

Even better – save **£1** per issue if you subscribe for two years – a total saving of **£24**.

Overseas rates represent exceptional value

- You also:
- Avoid any cover price increase for the duration of your subscription
 - Get your magazine delivered to your door each month
 - Guarantee a copy, even if the newsagents sell out

- Order by:
- Phone or post with a cheque, postal order or credit card
 - Or order online at: www.epemag.com (go to our secure server-backed online shop)



SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in:

UK: 6-months £26.99, 12-months £49.85, 24-months £94.99

Europe Airmail: 6-months £30.99, 12-months £57.99, 24-months £109.99

Rest Of The World Airmail: 6-months £37.99, 12-months £70.99, 24-months £135.99

Cheques (in **£ sterling only**) payable to *Practical Electronics* and sent to: Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom

Tel: 01202 087631

Email: pesubs@selectps.com

Also via our secure online shop at: www.epemag.com

Subscriptions start with the next available issue. We accept MasterCard or Visa.

(For past issues see the Back Issues page.)



DIGITAL SUBSCRIPTIONS

Electronic subscriptions (PDF download or online) are available from £20.99 for 12 months, more details at: www.epemag.com

SUBSCRIPTION ORDER FORM

- ☐ 6 Months: UK £26.99, Europe £30.99 (Airmail), Rest of the World £37.99 (Airmail)
- ☐ 1 Year: UK £49.85, Europe £57.99 (Airmail), Rest of the World £70.99 (Airmail)
- ☐ 2 Years: UK £94.99, Europe £109.99 (Airmail), Rest of the World £135.99 (Airmail)

To: Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH
United Kingdom

Tel: 01202 087631

Email: pesubs@selectps.com

I enclose payment of £
(cheque/PO in £ sterling only)
payable to *Practical Electronics*

☐ Please charge my Visa/Mastercard



My card number is:
Please print clearly, and check that you have the number correct

Card Security Code Valid From Date.....
(The last 3 digits on or just under the signature strip)

Card Ex. Date

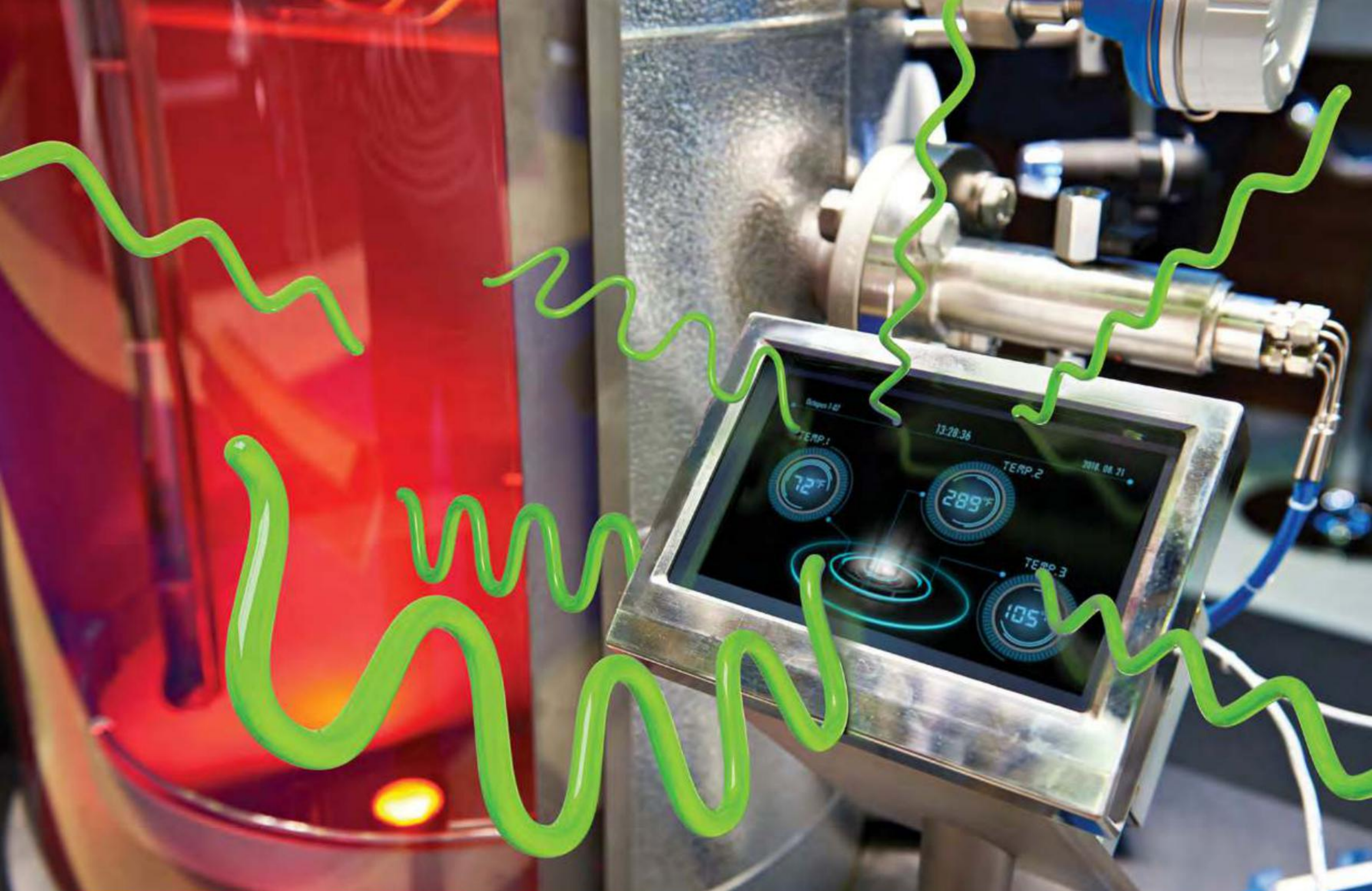
Name

Address

Post code Tel.

Email

Subscriptions start with the next available issue.



Measure the World Around You

Solutions to Condition Every Signal

Bridging the gap between the world we live in and the digital world is a complex task. Microchip's standalone mixed-signal, linear and interface products enable designers to easily and accurately collect, measure, monitor and convert a wide variety of signals to improve your product designs. In addition to a wide variety of general-use products, we also have a selection of high-performance devices to help you meet your strict design requirements.

Explore our options to condition your signals!



Explore your options at www.microchip.com/Signal



WIRELESS FOR THE WARRIOR

by LOUIS MEULSTEE

THE DEFINITIVE TECHNICAL HISTORY OF RADIO COMMUNICATION EQUIPMENT IN THE BRITISH ARMY

The *Wireless for the Warrior* books are a source of reference for the history and development of radio communication equipment used by the British Army from the very early days of wireless up to the 1960s.

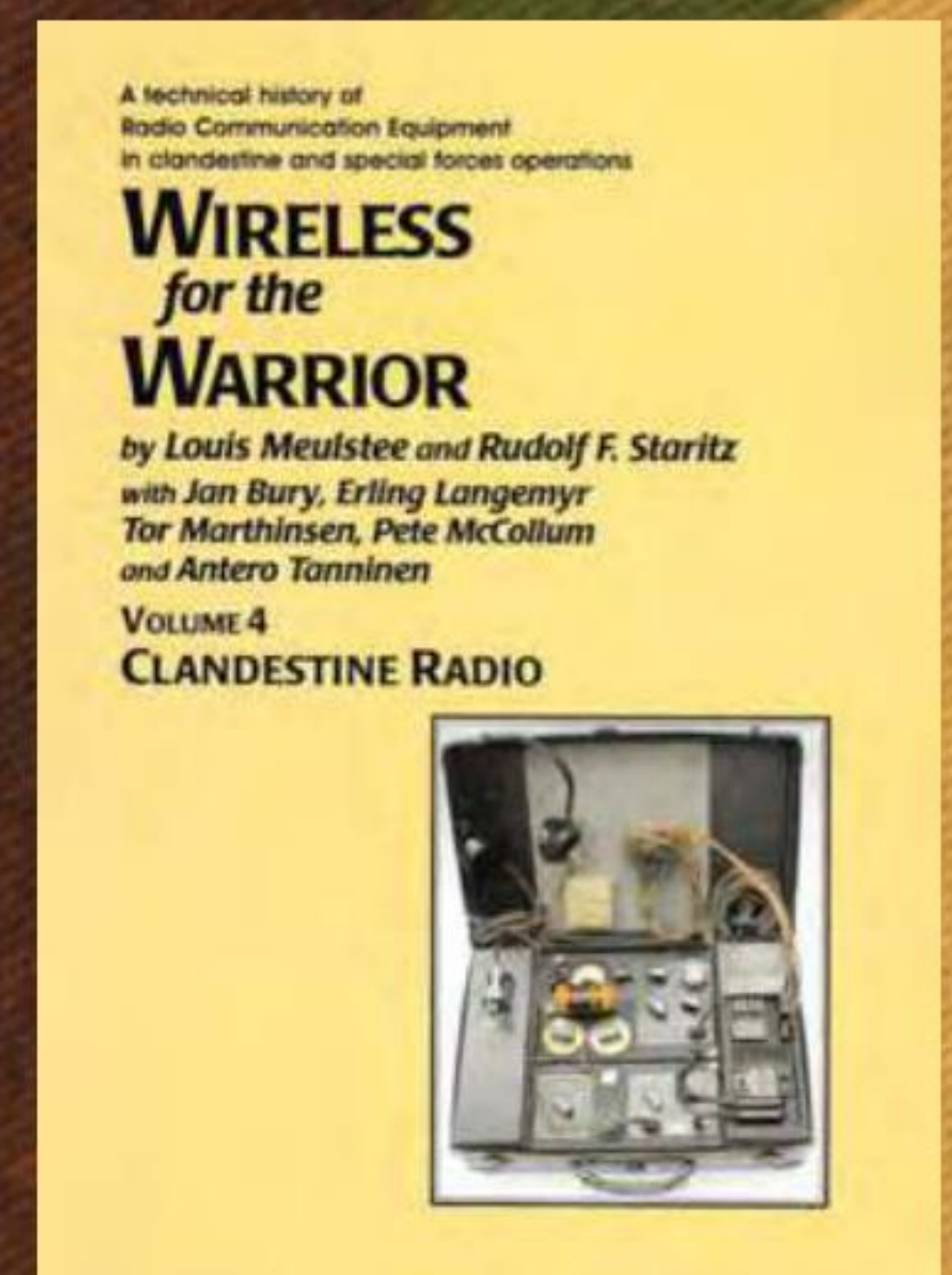
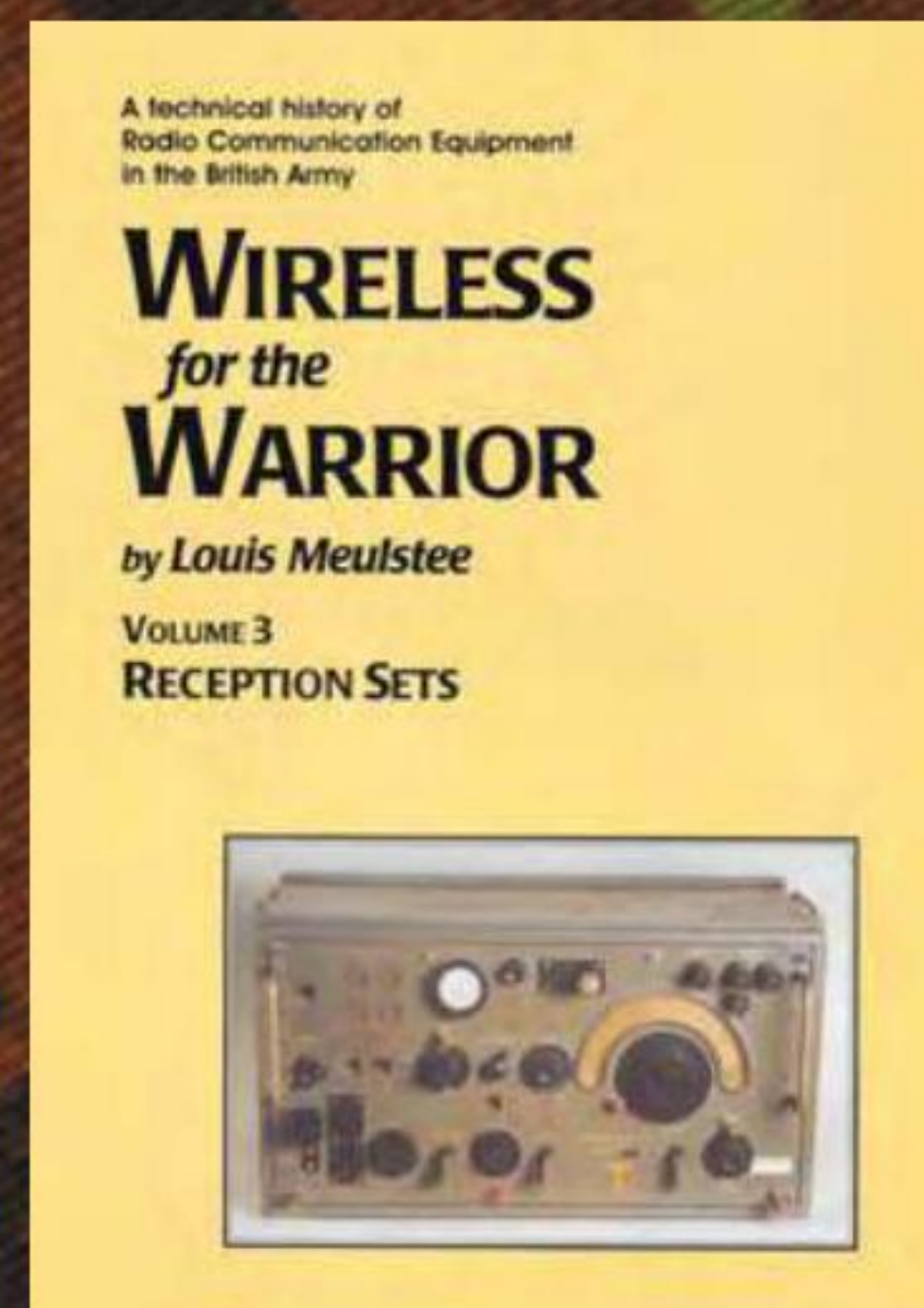
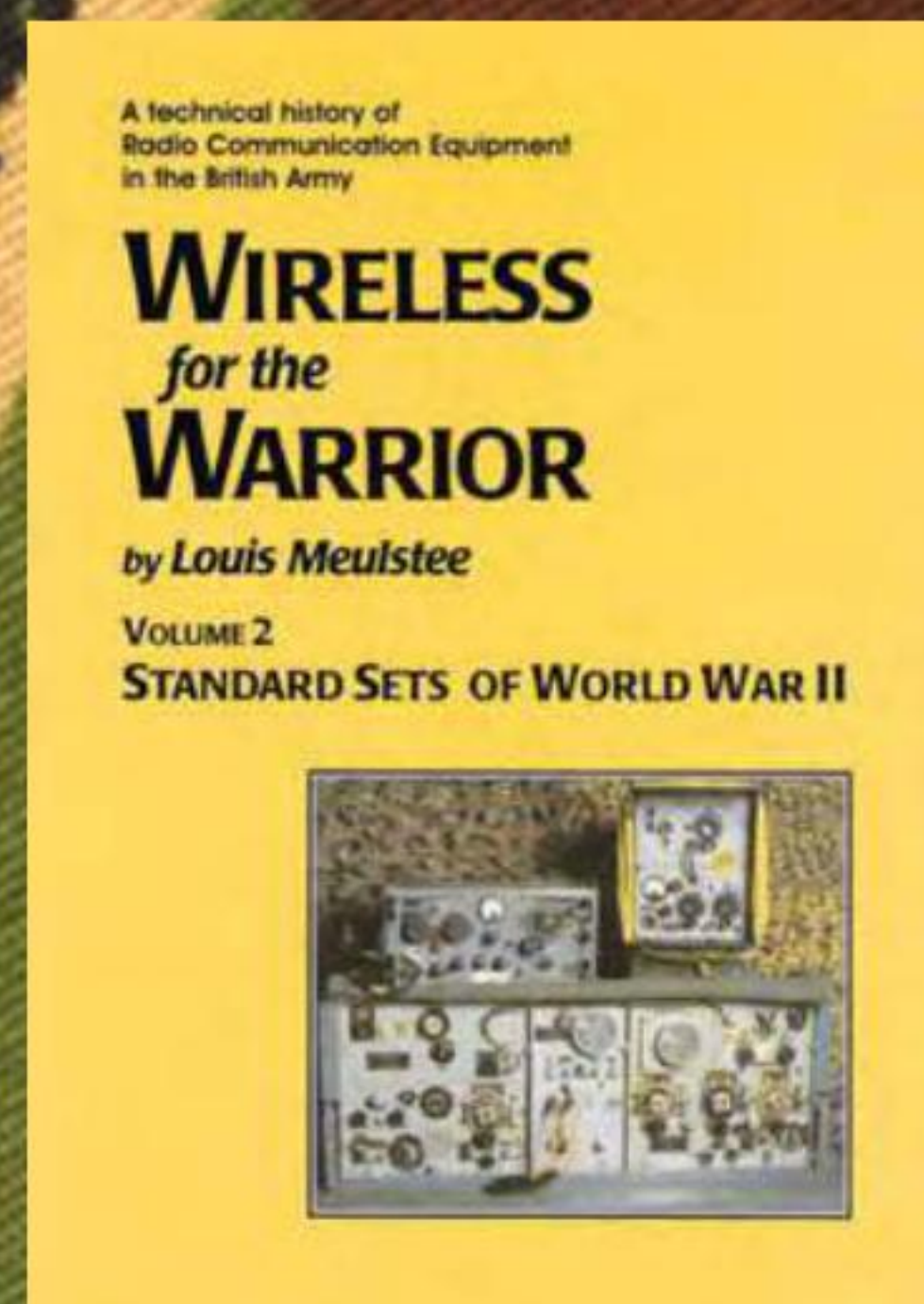
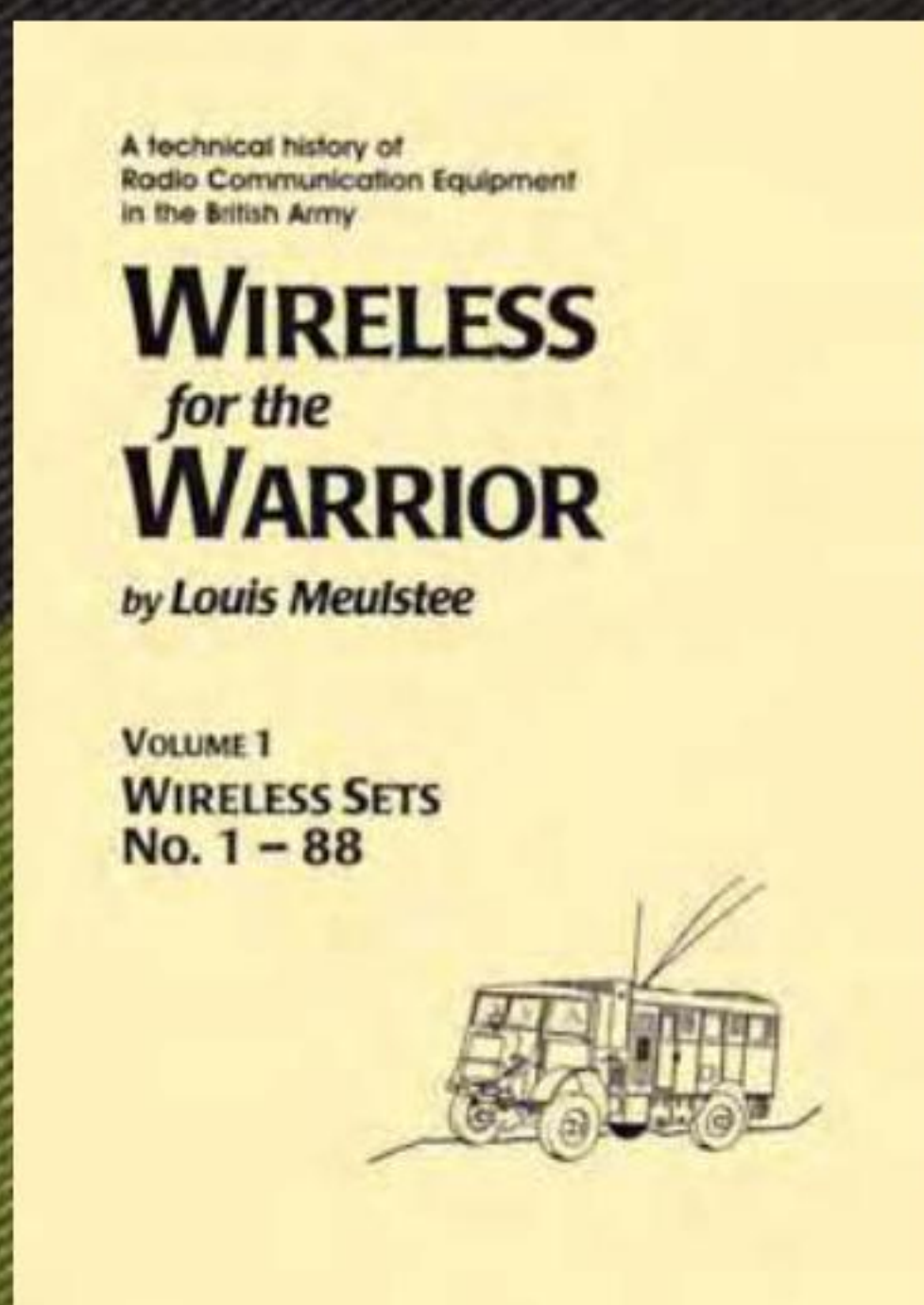
The books are very detailed and include circuit diagrams, technical specifications and alignment data, technical development history, complete station lists and vehicle fitting instructions.

Volume 1 and *Volume 2* cover transmitters and transceivers used between 1932-1948. An era that starts with positive steps taken to formulate and develop a new series of wireless sets that offered great improvements over obsolete World War I pattern equipment. The other end of this

timeframe saw the introduction of VHF FM and hermetically sealed equipment.

Volume 3 covers army receivers from 1932 to the late 1960s. The book not only describes receivers specifically designed for the British Army, but also the Royal Navy and RAF. Also covered: special receivers, direction finding receivers, Canadian and Australian Army receivers, commercial receivers adopted by the Army, and Army Welfare broadcast receivers.

Volume 4 covers clandestine, agent or 'spy' radio equipment, sets which were used by special forces, partisans, resistance, 'stay behind' organisations, Australian Coast Watchers and the diplomatic service. Plus, selected associated power sources, RDF and intercept receivers, bugs and radar beacons.



ORDER YOURS TODAY!
JUST CALL 01202 880299 OR VISIT www.epemag.com

Editorial offices

Practical Electronics
Electron Publishing Limited
1 Buckingham Road
Brighton
East Sussex BN1 3RA

Tel 01273 777619
Mob 07973 518682
Fax 01202 843233
Email pe@electronpublishing.com
Web www.epemag.com

Advertisement offices

Practical Electronics Adverts
1 Buckingham Road
Brighton
East Sussex BN1 3RA

Tel 01273 777619
Mob 07973 518682
Email pe@electronpublishing.com

Editor Matt Pulzer
General Manager Louisa Pulzer
Digital subscriptions Stewart Kearn **Tel** 01202 880299
Online Editor Alan Winstanley
Web Systems Kris Thain
Publisher Matt Pulzer

Print subscriptions

Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH
United Kingdom

Tel 01202 087631
Email pesubs@selectps.com

Technical enquiries

We regret technical enquiries cannot be answered over the telephone. We are unable to offer any advice on the use, purchase, repair or modification of commercial equipment or the incorporation or modification of designs published in the magazine. We cannot provide data or answer queries on articles or projects that are more than five years old.

Questions about articles or projects should be sent to the editor by email: pe@electronpublishing.com

Projects and circuits

All reasonable precautions are taken to ensure that the advice and data given to readers is reliable. We cannot, however, guarantee it and we cannot accept legal responsibility for it.

A number of projects and circuits published in *Practical Electronics* employ voltages that can be lethal. You should not build, test, modify or renovate any item of mains-powered equipment unless you fully understand the safety aspects involved and you use an RCD (GFCI) adaptor.

Component supplies

We do not supply electronic components or kits for building the projects featured, these can be supplied by advertisers. We advise readers to check that all parts are still available before commencing any project in a back-dated issue.

Advertisements

Although the proprietors and staff of *Practical Electronics* take reasonable precautions to protect the interests of readers by ensuring as far as practicable that advertisements are bona fide, the magazine and its publishers cannot give any undertakings in respect of statements or claims made by advertisers, whether these advertisements are printed as part of the magazine, or in inserts. The Publishers regret that under no circumstances will the magazine accept liability for non-receipt of goods ordered, or for late delivery, or for faults in manufacture.

Transmitters/bugs/telephone equipment

We advise readers that certain items of radio transmitting and telephone equipment which may be advertised in our pages cannot be legally used in the UK. Readers should check the law before buying any transmitting or telephone equipment, as a fine, confiscation of equipment and/or imprisonment can result from illegal use or ownership. The laws vary from country to country; readers should check local laws.

Has it really been a year?

It's hard to believe, but this *Editorial* marks the start of my second year as editor/publisher of *PE*. Where does the time go? I don't think we've ever been so busy, keeping what sometimes felt like an unlimited number of plates spinning. But, so far – touch wood – the sky hasn't fallen on our heads. The pagination is up (84 pages for every 2019 issue), we've had a complete redesign and of course the welcome return of our old/new title – *Practical Electronics*. Despite the hard work, it has been hugely enjoyable, and as Mike Kenward, our previous publisher said this time last year, it really is a pleasure to receive a finished magazine each month from our excellent printer (Acorn in Wakefield). Something tangible we've had a hand in creating.

Thank you!

Practical Electronics depends on two vital groups. First, our talented and dedicated writers and staff. So, to mark the end of 2019, and in no particular order – a heartfelt 'well done and thank you' to Alan Winstanley, Mike Tooley, Ian Bell, Mark Nelson, Mike Hibbett, Clive 'Max' Maxfield, Phil Boyce, Julian Edgar, Barry Fox and Jake Rothman. Plus, I'd like to offer an extra special 'thank you' to Stewart Kearn. He has provided unstinting and invaluable back-office support over the last 12 months as I attempted to learn the art of publishing a magazine. Last, but not least, I'd like to express my appreciation to Kris Thain who, along with Alan Winstanley, has been steadily building our new website.

The second group is just as important – you! Without our loyal readers and subscribers in the UK and across the globe *Practical Electronics* would cease to exist. Your continued support is truly appreciated by everyone at Electron Publishing.

Extra special Christmas present?

If you dread the inevitable 'What would you like for Christmas?' interrogation from family and friends, then why not suggest a subscription to your favourite magazine. It doesn't matter whether you choose paper or online, as a subscriber you can be sure that you won't miss out.

From all of us at *Practical Electronics*, have a very happy Christmas and a rewarding, silicon-filled 2020!

Matt Pulzer
Publisher

The Fox Report

Barry Fox's technology column



Peripheral vision at the movies

The cinema industry is always looking for new ways to lure punters away from their TV couches and into movie theatres where they can perhaps be persuaded to pay over the odds for sugary drinks and noisy smelly food. We have had bigger screens (IMAX), clearer pictures (4K laser projection), louder sound from more speakers in more places (Dolby Atmos) and 3D (now decidedly on the way out).

The latest enticement is 'ScreenX', which the Cineworld cinema at the O2 Centre in Greenwich recently unveiled with the racing car movie, *Le Mans '66*.

To mark the launch, a panel of 'experts' pontificated: Jon Bentley, presenter of *The Gadget Show*, professor Sarah Atkinson, senior lecturer at King's College London, immersive technologies and film expert, and Phil Peirce, Cineworld Cinema Operations.

In addition to the main screen digital laser projector, four additional 4K projectors (two on each side of the audience) beamed images onto the cinema sidewalls. In theory, all five images stitch together to create a 270° immersive experience, like a modern version of the old Cinerama film system.

Phil Peirce assured that not all movies would use ScreenX – and ScreenX movies would not use ScreenX all the time; only for sequences that benefit from peripheral-vision content. Although existing movies can be 'converted' to ScreenX, he said, *Le Mans '66* is the first to be shot with extra cameras capturing

side content. The Cineworld chain, he went on, already has 15 ScreenX cinemas and is committed to 100. The format is ideal for live music events, he suggested.

Prof Atkinson thought ScreenX was 'another way to tell stories and an added incentive to go the cinema'.

Most people, I suspect, will like what the peripheral vision effect brings to the race sequences in *Le Mans '66*. So as long as cinema owners do not overuse the sidewall projectors, ScreenX could perhaps be a useful addition to the existing armoury of cinema special effects.

The downside is that although the side images overlap and stitch fairly neatly, there is an obvious black gap between the side images and main front screen image. Worse, because the four side projectors are left 'on' all the time, they make the side walls glow grey when not being used for movie content. This creates light spill, which adversely affects the main screen contrast. (And the cinema's illuminated Exit signs become part of the side pictures.)

For the *Le Mans '66* screening there were no deep blacks in either the side or main image. We hear a lot about HDR, High Dynamic Range; this was decidedly LDR, Low Dynamic Range. Also, the main picture was very soft, as if quite seriously out of focus. It looked more like poor 35mm film, or even 16mm film. Why?



270° cinema – the next big thing, or just a gimmick?

The panel members said they'd be 'around afterwards to talk', but they all left before the end. So the next day I messaged Cineworld, asking Phil Peirce for comment. Surely Cineworld would check focus during a screening, and surely 20th Century Fox would not supply a bad master copy for public screening?

Over the next week I reminded several times and several times I was promised comment. Finally, Phil Peirce ducked the issue and Cineworld issued bland corporate waffle: 'Cineworld prides itself on screening films using state-of-the-art technology creating the most immersive and atmospheric experience for cinemagoers. We regularly check the quality of our sound and picture to ensure that this is the case, including on the day of the screening of *Le Mans '66*. The overall feedback we have received on the screening has been overwhelmingly positive.'

I'm beginning to suspect that ScreenX will disappear from the movie scene even faster than 3D.

Barry Fox, FBKS (Fellow, International Moving Image Society)



**Die-cast enclosures
standard & painted**

www.hammondmfg.com/dwg.htm

www.hammondmfg.com/dwg_SBVer.htm

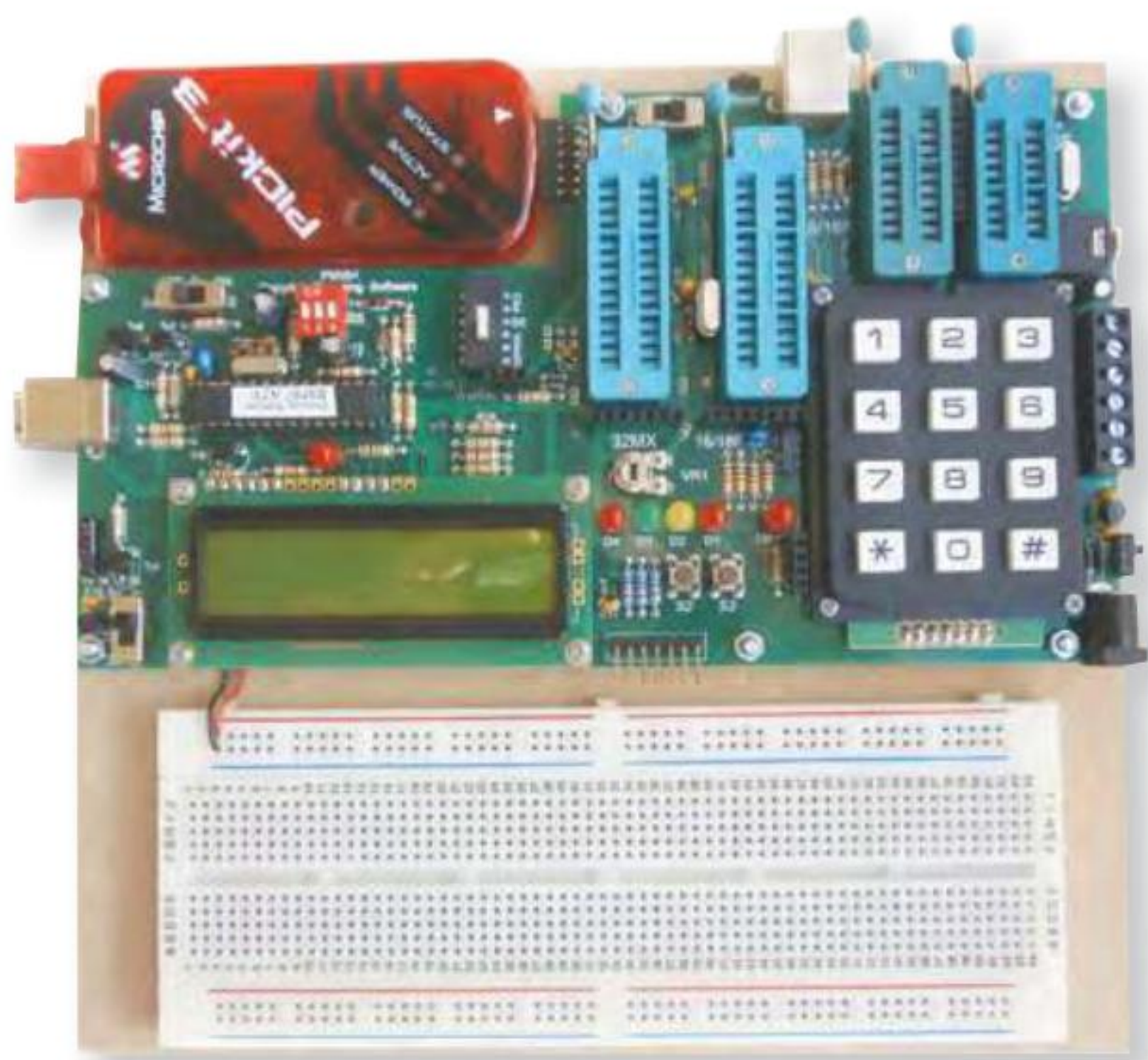
01256 812812

sales@hammond-electronics.co.uk



P955H PIC Training Circuit

by Peter Brunning



When you are first learning about PICs, whether you are a complete beginner or an experienced programmer, you need an uncomplicated system which allows you to learn without getting bogged down in system procedure. That is why we created the P955H PIC training circuit and our own PIC assembler. In the first book we learn about PIC programming using the Brunning Software PIC assembler BSPWA, but in chapter 3M there is an introduction to the Microchip assembler MPASM X. All our assembler text will run in both systems, so from there on, if you wish you can use MPASM X. Likewise, we start by using the on-board PIC programmer to write the code into the PIC, but if you prefer, plug in a PICKit 3 and use that. The P955H training circuit has the flexibility to be what you need as your learning process advances.

The P955H training circuit has been designed to work with both 32-bit and 8-bit PICs. The idea is to start learning about PICs using assembler with 8-bit PICs. Then learn C with 8-bit PICs, study serial communications using 8-bit PICs, and finally study C programming using 32-bit PICs. It is a simple approach to a subject that has no limit to its ultimate complexity.

The Brunning Software P955H PIC Training Course

We start by learning to use a relatively simple 8-bit PIC microcontroller. We make our connections directly to the input and output pins of the chip and have full control over the internal facilities of the chip. We work at the grassroots level.

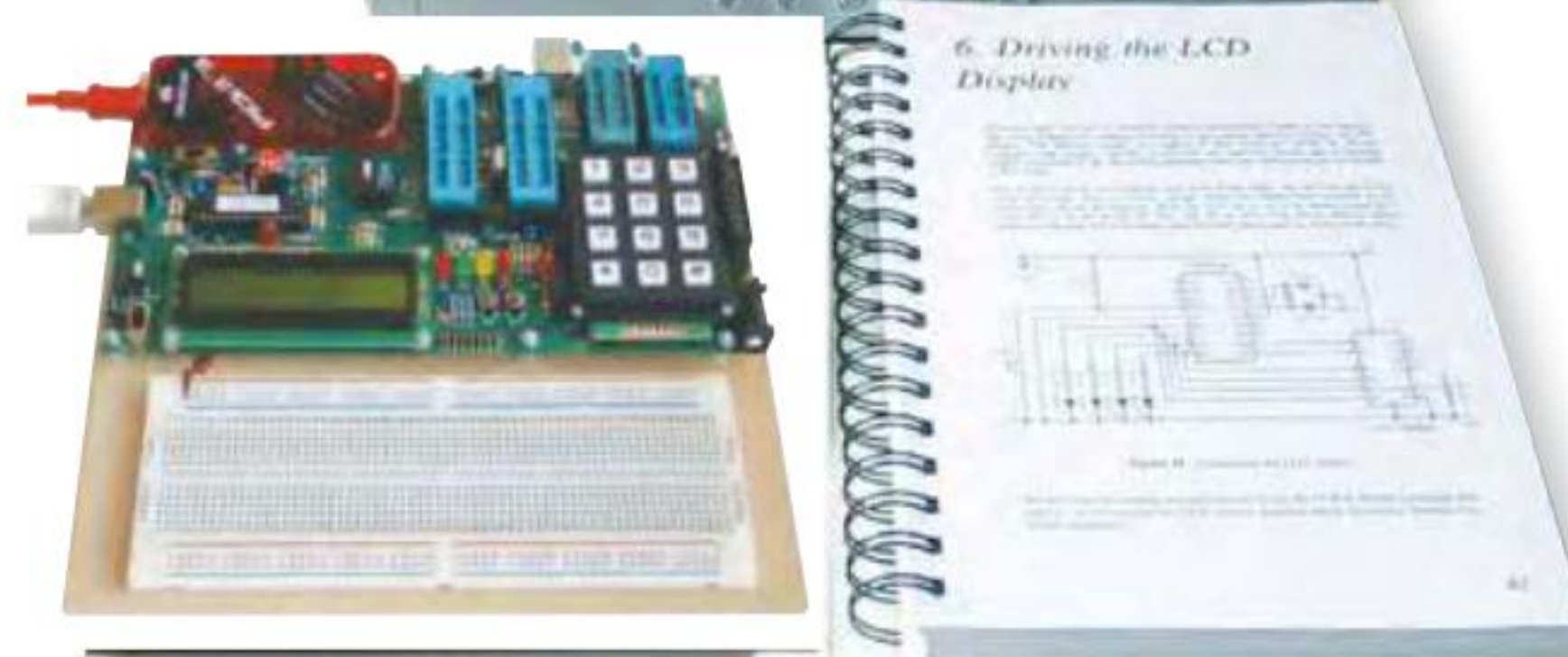
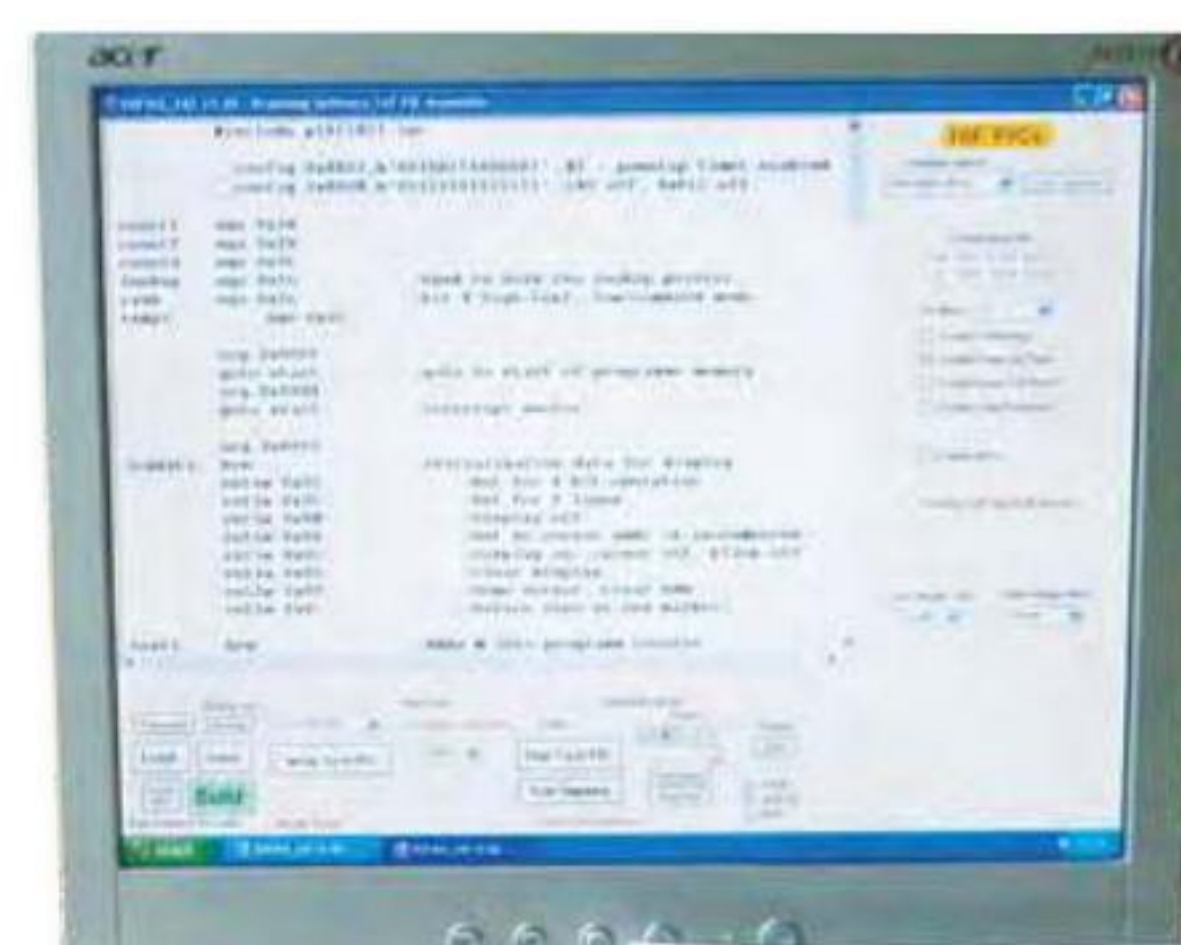
The first book teaches absolute beginners to write PIC programmes using assembler, which is the natural language of the PIC. The first book starts by assuming you know nothing about PICs, but instead of wading into the theory we jump straight in with four easy experiments. Then, having gained some experience, we study the basic principles of PIC programming, learn about the 8-bit timer, how to drive the alphanumeric liquid crystal display, create a real-time clock, and experiment with the watchdog timer, sleep mode, beeps and music. Then there are two projects to work through. In the space of 24 experiments, two project and 56 exercises we work through from absolute beginner to experienced engineer level using the latest 8-bit PICs (16F and 18F).

The second book introduces the C programming language for 8-bit PICs in very simple terms. The third book, *Experimenting with Serial Communications*, teaches Visual C# programming for the PC so that we can create PC programmes to control PIC circuits.

In the fourth book, we learn to programme 32-bit MX PICs using fundamental C instructions. Flash the LEDs, study the 16-bit and 32-bit timers, write text to the LCD, and enter numbers using the keypad. This is all quite straightforward as most of the code is the same as already used with the 8-bit PICs. Then life gets more complex as we delve into serial communications with the final task being to create an audio oscilloscope with advanced triggering and adjustable scan rate.

The complete P955H training course is £259, which includes the P955H training circuit, four books (240 × 170mm, 1200 pages total), six PIC microcontrollers, PIC assembler and programme text on CD, two USB-to-PC leads, a pack of components, and carriage to a UK address. (To programme 32-bit PICs you will need to plug in a PICKit 3, which you can buy from Microchip for £38.)

Prices start from £175 for the P955H training circuit with Books 1 and 2 (240 × 170mm, 624 pages total), two PIC microcontrollers, PIC assembler and programme text on CD, USB-to-PC lead, and carriage to UK address. (PICKit 3 not needed for this option). You can buy Books 3 and 4, USB PIC, 32-bit PIC and the components kit as required later. See the Brunning website for details: www.brunningsoftware.co.uk



Mail order address:

Brunning Software

138 The Street, Little Clacton, Clacton-on-sea,
Essex, CO16 9LS. Tel 01255 862308

New year, new start, so for at least one month this column reverts to a rant-free zone, concentrating on more or less practical electronics. But that won't stop us from exposing pseudo-technical mumbo-jumbo and crackpot theories. Let's begin with a little eye-rolling at some weird nonsense dressed up as science.

Ever heard of shungite? Me neither; but you will soon because it's being touted as the new miracle remedy for a wide range of health problems including brain and heart tumours, DNA breakage, chronic fatigue and much more. All allegedly caused by 'electronic smog'. Shungite, as Wikipedia explains helpfully, is a black, lustrous, non-crystalline mineraloid consisting of more than 98% carbon. You can buy shungite health bracelets, water filter crystals or even a 20mm disc of shungite that 'simply fits into any smartphone case to expel radiation from 3G, 4G and 5G rays'. You can see how effectively the disc works on YouTube at: <https://youtu.be/I9SfCYjKp5g> and, as we all know, the camera cannot lie. An even more scientific demonstration can be watched at: <https://youtu.be/BQ9jOAwC7JY>

Shungite even neutralises AIDS, according to this guy – as long as it's the expensive 'noble' variety of snake oil, I mean shungite. Seriously, how is this nonsense delivered with a straight face? If you feel like reading more pseudo-science or just want a good laugh, visit: <http://bit.ly/pe-jan20-tt1>

Impractical electronics

Once again, life imitates art. First, we thought that lead acid cells were the only practical type of rechargeable battery, then along came NiCad technology, followed by nickel-metal-hydride cells, lithium-ion and the most recently, a new contender – lithium-ion-polymer batteries. Each successive type aims to surpass its predecessors in effectiveness, power-density or cost.

Not to be outdone, Mother Nature has just revealed two 'new' types of electric eel, one packs a powerful punch of 650-860V. In truth, these types (or 'species' as they should be called) have been around for some time, but they have only just been discovered. Researchers working in South America, where all electric eels are found, have identified two new species, one of which (*Electrophorus voltai*) is cited as the world's strongest living bioelectricity generator.

To measure the peak power output of these shocking creatures, each subject specimen was stretched out on a dry heavy-duty (non-conductive) plastic sheet to isolate it from the resistive loading of water. In this position, a DC-coupled voltage reading from snout to the distal end of the tail was taken by gently prodding the tip of the snout to elicit a volley of high-voltage discharges, measured with a Fluke 190-202 storage oscilloscope. The entire procedure was accomplished in less than one minute.

So far, nobody appears to have put electric eels to use as a power supply, nor discussed the ethics of so doing, so we must reluctantly file this information under 'impractical electronics'.

Cool magic

Far more practical is a recent development in free energy, the invention of a \$30 (£23) device that converts temperature difference into electricity. Given that it produces only enough energy to light one LED, you might call this pretty inconsequential, but that's not what its inventors think. And given that it works at night and consequently does not rely on solar energy, they argue that their approach is immediately practical for lighting and off-grid sensors.

'A large fraction of the world's population still lacks access to electricity, particularly at night when photovoltaic systems no longer operate. The ability to generate electricity at night could be a fundamentally enabling capability for a wide range of applications, including lighting and low-power sensors,' argue Prof. Aaswath Raman and his colleagues at Stanford University and the University of California. They explain that their device works by exploiting the effect of radiative sky cooling, in which all sky-facing surfaces become colder than the surrounding air as they naturally radiate heat into the atmosphere at night when the sky is cool. The difference in temperature between the sky-facing surface and its underside can be exploited to generate electricity using a low-cost thermoelectric generator.

The researchers explain: 'Unlike traditional thermoelectric generators, our device couples the cold side of the thermoelectric module to a sky-facing surface that radiates heat to the cold of space and has its warm side heated by the surrounding air, enabling electricity generation at night. Experimentally we have demonstrated 25mW/m² of power generation and validated a model that accurately captures the device's performance, thereby generating light from the darkness of space itself.'

How it works

To make this device, start with a low-cost COTS (commercial off the shelf) thermoelectric module. Then attach one side of it to a 20cm-diameter aluminium disc that has been painted black (with normal paint). Then you put this in a plastic enclosure covered with mylar and place it under a transparent cover.

The black-disc side of the box faces the sky and radiates heat out into space, meaning it is slightly colder than ambient air. The other side of the box carries another 20cm aluminium disc that is not painted and is warmed by the surrounding air. At night, this results in a temperature difference of a few degrees Celsius, as the upper surface is cooler than the bottom of the generator. When tested at night in California, under clear skies, the prototype generator produced up to 25mW/m². However, in a Mediterranean or desert climate, with warm, dry conditions, the developers are confident their device would generate half a watt per square metre.

Bright future?

Raman predicts that since the generator is simple and made with low-cost commonly available materials, it should be easy to produce on a large scale. Further engineering improvements may include better insulation or increasing the area of the heat radiator. He sums up: 'Beyond lighting, we believe this could be a broadly enabling approach to power generation suitable for remote locations, and anywhere where power generation at night is needed.'

Win a Microchip MPLAB PICkit 4 In-Circuit Debugger

Practical Electronics is offering its readers the chance to win a Microchip MPLAB PICkit 4 In-Circuit Debugger (PG164140).

The Microchip MPLAB PICkit 4 In-Circuit Debugger allows fast and easy debugging and programming of PIC and dsPIC Flash microcontrollers, using the powerful graphical user interface of the MPLAB X integrated development environment (IDE).

The MPLAB PICkit 4 programs faster than its predecessor thanks to a powerful 32-bit 300MHz SAME70 MCU, and comes ready to support PIC and dsPIC MCU devices. Along with a wider target voltage, PICkit 4 supports advanced interfaces such as 4-wire JTAG and Serial Wire Debug with streaming Data Gateway, while being backward compatible for demo boards, headers and target systems using 2-wire JTAG and ICSP.

**Worth
\$47.95
(approx £38)
each**

Key features of the PICkit 4 include matching silicon clocking speed, supplying up to 50mA of current to the target, a minimal current consumption at <100µA from target, as well as an option to be self-powered from the target.

The MPLAB PICkit 4 is connected to the design engineer's computer using a high-speed USB 2.0 interface and can be connected to the target via an 8-pin single in-line (SIL) connector. The connector uses two device I/O pins and the reset line to implement in-circuit debugging and in-circuit serial programming (ICSP).

Currently, the MPLAB PICkit 4 In-Circuit Debugger/Programmer supports many but not all PIC MCUs and dsPIC DSCs, but is being continually upgraded to add support for new devices.

Free-to-enter competition



Microchip MPLAB PICkit 4 In-Circuit Debugger

How to enter

For your chance to win a Microchip MPLAB PICkit 4 In-Circuit Debugger, enter your details in the online entry form at:

<http://page.microchip.com/PE-PICkit-4.html>

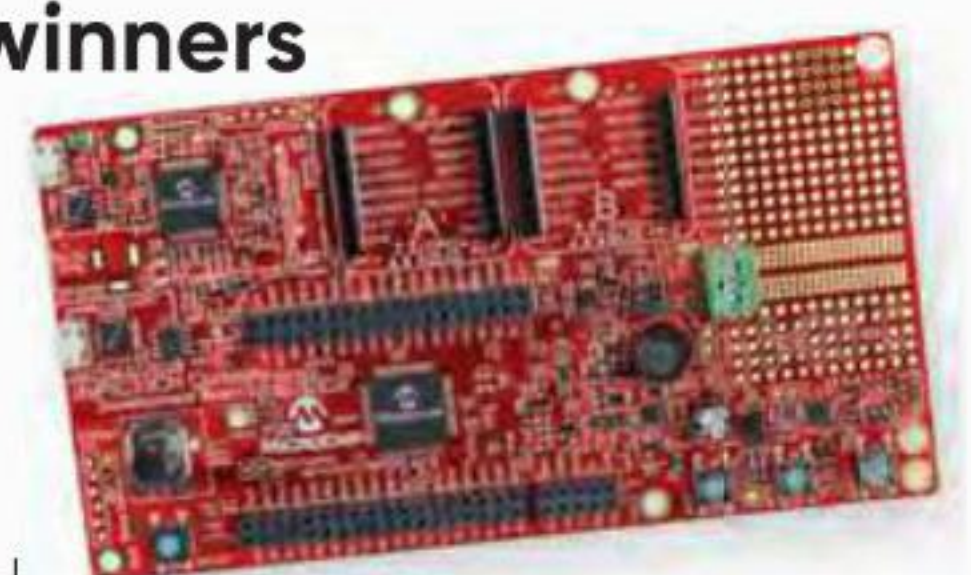
Closing date

The closing date for this offer is 31 January 2020.

September 2019 winners

Ed Rees
Michael Howell

Each won a
Microchip dsPIC33CH
Curiosity Development Board worth £27.50



Net Work

Alan Winstanley

This month, a reminder of the single most important law of computing. Back up now, because there are only two kinds of computer – ones that have failed, and ones that are going to fail.

This month's column nearly didn't see the light of day, as the author's main PC decided to self-destruct inconveniently one Friday afternoon. It seems a Windows update failed to install correctly when the PC was left unattended, and the author returned only to find that the Windows OS had corrupted itself. Dire pop-ups warned of 'catastrophic errors' and I was left with no choice – the machine would need a re-build. Although *Net Work* isn't a computer column, this month's article starts with a timely reminder about thinking ahead and avoiding data loss should unforeseen problems strike.

Solid foundations for a new PC

Fortunately, the PC still booted into Safe Mode, and the data files were intact. I opted to re-install the operating system onto a solid-state disk, and a Samsung 500GB 860 EVO SSD arrived next morning. Windows was re-installed as normal, followed by the usual litany of expansion cards, drives, peripherals and software.

I decided to retain the old hard drive for accessing the bulk of existing data. The legacy Windows 'Libraries' feature (hidden by default in Windows 10) creates shortcuts to key document and data folders stored elsewhere, and new ones can easily be added (right-click on target folders and go Include in library, or Create new library), which made linking to the legacy data an easy job. This arrangement works well, though changing one's old navigational habits takes some getting used to.



An SSD upgrade offers several benefits over its mechanical counterpart. (Photo: Samsung)

Data, bookmarks and history files from old installations of, for example, Firefox can also be ported over if you know where to look (Google offers a wealth of advice). Some software was also treated to a long overdue upgrade, notably the indispensable Mailwasher by Firetrust (www.mailwasher.net), which screens junk POP3 mail from multiple mailboxes.

As computer users know all too well, there's always a risk that a hard disk, motherboard or even an SSD could fail without warning (Samsung 'Magician' software lets you monitor the SSD's health) and life is made easier by planning for problems before a PC has an untimely meltdown. Data can be stored on free or paid-for cloud storage, including Microsoft OneDrive and Google Drive, but portable SSDs are catching on too. The author uses a RAID NAS for data safe-keeping and, being a practical man, hard copies of major software serial numbers (eg invoices) are also filed safely away, while driver disks and corresponding manuals are bunched up into A5 plastic wallets. This at least helps get things going again during a crisis-driven rebuild.

Finally, on our computer topic, a reminder to Windows 7 users that Microsoft is ceasing extended support on 14 January 2020, after which no new security patches will be made available. Plenty of users, including the writer, still prefer the Windows 7 GUI over the flat tiles of Windows 10, but we will doubtless be forced inexorably into adopting the latter's 'Metro' style some time in the future.

Fully charged

In last month's *Net Work*, UK power generation trends were examined and it showed how UK consumer demand has risen at a rate of about 1TWh (terawatt-hours) per year since 1970. Only now does it seem to be levelling off, according to BEIS statistics.

British consumers are being 'encouraged' – if not strong-armed – into installing smart meters, a campaign that has been met with scepticism and resistance by residential users. In my view,

there is a certain deceit by omission in the way that the smart meter programme has been promoted. Soft benefits, such as 'No more estimated bills' and 'No need to submit meter readings' are often cited; but in truth, smart meters are akin to the power sector sticking ECG electrodes on every rooftop, enabling providers to monitor power consumption and shape consumer behaviour for ever more.

The purpose of demand-side response (DSR – see last month) is to help match supply with demand, possibly throttling demand at peak times by electronically messaging consumers to 'request' a reduction in usage demand. Potentially, we might see some new imaginative pricing structures; for example, users could be penalised for using power on cold or calm days when demand was higher but less wind or renewable power (say) was available. There is no stopping the smart meter programme, but it's worth realising that what's really happening is that real-time technology is being put in place that can force consumers to change their power consumption dependent on weather, time of day or the availability of renewable energy, and bill them accordingly.

The appliance of science

In the 'smart' new home, a Smart Meter Home Area Network (SMHAN) using the Zigbee Smart Energy Standard will alter the way householders use some electrical appliances. Compatible equipment, such as dishwashers, washing machines or electric vehicle (EV) charging points could access pricing data via the SMHAN or through a home hub gateway device, and operate when cheaper energy rates prevail. This way, our homes really will get to know when power is cheapest or more readily available. In Britain, 'Economy 7' and 'Economy 10' tariffs offer cheaper off-peak rates for households that are blighted by electric storage heaters. So-called 'ALCS' (auxiliary load control switch) smart meters will therefore allow a separate 'auxiliary' circuit, such as an EV charger or heat pump to run at cheaper times of day or night.

With all the technology and network coverage needed by the smart meter programme, the astronomically expensive (£13bn) rollout of gas and electric smart metering equipment (GSME and ESME) has been hampered by lacklustre consumer demand and suspicions about privacy and security. Coupled with fundamental technology problems, the smart meter roll-out is set to drag on for at least another five years in Britain.

The first iteration of *Smart Metering Equipment Technical Specifications (SMETS1)* was signed off in 2012 and updated in 2014. The first wave of customers complained of poor connectivity, malfunctioning or meters going ‘dumb’. One user was charged £33,000 for a day’s electricity. Some users say that their IHD (in-home displays) ended up in a drawer once the novelty had worn off. A second standard (SMETS2) is intended to overcome earlier snags, as well as introducing more features. Complaints have now surfaced of suppliers converting some smart meters to ‘prepayment mode’ without consent.

The non-profit organisation that manages smart meter datacomms, the Data Communications Company, has been working to fix problems with SMETS1 meters and make them operable on the network. Some two million SMETS2 meters had been installed by September this year, and 10,000 a day are being fitted. Once smart meters are universal, expect to see radical changes in energy pricing structures in years to come: tomorrow’s utility bills could look very different from those we see today.

Grid lock

The ‘grid’ or smart meter wide area network (SM WAN) now covers 99.25% of consumers in northern Britain and 97.7% in the rest of the country, with



Awash with technology: in the future, smart appliances could ‘know’ when energy rates are lowest.

the contracted near full-coverage target to be reached by the end of 2021, they hope. The remaining 0.75% will remain ‘out of scope’ or beyond reach. In Britain, both the age and the location of properties determine the type of construction seen in domestic housing. This has affected the rollout of smart meters because centuries-old properties can have much thicker walls, and modern energy-saving building techniques have increased the uptake of metallised insulation. ‘K-glass’-type low-emission double glazing and foil-lined roofing materials have contributed to ‘building penetration loss’ which screens out the smart meter WAN signal. A new dual-band hub is being developed for use in blocks of flats or traditional British homes that might have very thick stone walls, or locations where the SM WAN signal might not otherwise reach. Field trials conducted with the Building Research Establishment are designed to yield new solutions to implement smart meter network coverage in coming years.

Contrast the troubled domestic smart meter programme with Britain’s wind power success story, which continues to expand predominantly in the hostile environment of the North Sea. Hornsea One (see *Net Work*, September 2019) currently claims the crown for being the world’s largest (by area) offshore wind farm, situated 120km off Britain’s coastline, with 174 7MW turbines generating 1.2GW. A new Hornsea Two wind farm will be closer to home, supplying a further 1.4GW using 165 8MW turbines. A joint venture between SSE and Norway’s Equinor (formerly Statoil) aims to construct three more offshore wind farms in the Dogger Bank area, generating 3.6GW in total (see www.doggerbank.com). They will be populated with GE Renewable’s new Haliade-X 12MW wind turbines, the most powerful offshore turbines yet developed.

Another ‘green’ problem is being stored up for the future though, as it has been discovered that fibreglass fan blades are difficult to recycle due to their sheer size and strength. In the US, *Wyoming News Now* reported that although 90% of a wind turbine can be recycled, 900 massive fan blades from defunct turbines were being sliced up and sent to landfill.



The new 12MW GE Renewable Energy Haliade-X claims to be the most powerful offshore wind turbine available.

And finally

As mentioned in the June 2019 column, after acquiring Eero, the US creators of low-cost mesh networking, Amazon has launched its own mesh Wi-Fi range offering seamless network coverage around the home. Each Eero router costs £99 and the tri-band Eero Pro is £179 each. Multi-packs are also available. A 4th generation Echo Dot smart speaker launched by Amazon includes an LED digital clock.

Christmas shoppers are being reminded not to be swayed by fake or paid-for reviews that promote dodgy goods sold on popular websites. There is also no guarantee that a seller is UK-based either, as goods may ship direct from China in a matter of days. Britain’s Which? consumer body investigated some tricks used by unscrupulous Amazon sellers earlier this year, see: <http://bit.ly/pe-jan20-nw>

Current-generation Huawei mobile phones are still receiving updates for Google apps and Android despite an aggressive US trade ban (see *Net Work*, August 2019). The future for new products is less certain, but Huawei is floating its new Harmony OS, which it hopes might replace stock Android in the future. Last October, BT CEO Philip Jansen claimed it would take seven years to remove Huawei networking equipment from existing UK infrastructure [if it proved necessary].

The US tech store Best Buy has ceased supporting its own-brand Insignia smart device range, leaving owners of their products stranded with ‘dumb’ appliances. Users of Insignia Wi-Fi cameras are especially aggrieved, and some customers have been offered vouchers as compensation.

Hard disk permitting, see you next month for more *Net Work*.

The author can be reached at:
alan@epemag.net

Design
Phil
Prosser
Words
Nicholas
Vinen



Audio DSP

Part 1

This is an exciting project, but what exactly is it? A digital signal processor? – a two-way active crossover? – or perhaps an 8-channel parametric equaliser? In fact, it is all of these... and more.

There's a wide range of audio processing tasks this project can handle. It uses DSP to provide an 8-channel parametric equaliser, so you can adjust frequency response to exactly the way you want it with really low distortion and noise. Or you can use it to 'bi-amplify' a pair of speakers. Or you can simply use it to experiment with any audio signal. Best of all, its modular design means it's ready for future expansion.

Let's face it: most tone controls don't give you a huge amount of control. Sure, you can boost or cut the treble and bass – but only centred on particular frequencies. Yes, you can adjust the level between channels – but that's about it. Wouldn't you like to have total control over your sound system? If so, you need this *Active Crossover/DSP/Parametric Equaliser*. It simply slots in between your sound source (no preamp required) and your amplifier (if your amp has tone controls, simply leave them 'flat').

We've published active crossovers before (the latest in September/October 2018), and DSP-based projects before (October 2015), but this is the first time we've combined both concepts. This is also the first time that we're publishing a digital signal processor that's truly high fidelity; it has a very low THD (total harmonic distortion) figure of around 0.001%.

This unit takes a stereo audio signal and splits it into two separate audio signals, with two output channels containing only the high frequencies and the other two, the low frequencies. These can then be fed to separate stereo amplifiers, with one amplifier driving the tweeters and the other driving the woofers. The signals combine in the air to give an accurate reproduction of the original audio signal.

This avoids the need for passive crossover circuitry, which can reduce sound quality, and allows for higher total power output, due to each amplifier only having to handle part

of the audio signal. It can be tweaked to perfectly suit the drivers and cabinet used, as DSP allows for the crossover parameters to be set precisely and identically between the left and right channels.

Since the chip is already processing the digital audio data, we've also provided some parametric equalisation, so you can modify the frequency response of the unit to compensate for any deficiencies in your drivers, cabinet, placement, room and so on.

Basically, you can tweak the sound profile to be exactly the way you like it, and without any further degradation to the audio signal, since it's only converted from analogue to digital and back to analogue once, no matter how much additional processing is done in the digital domain.

What the Active Crossover does

Fig.1 describes what the unit does. It shows the spectrum of an audio signal, with the frequency increasing left-to-right, from the lowest frequency that we can hear to the highest. The level of each component of this signal is shown in the vertical axis.

The blue area shows the signals which are extracted from the input to be sent onto the tweeter, while the mauve area shows those which go to the woofer. Signal components which fall in the crossover zone in the middle go to both outputs, although at reduced levels, so that they add up in such a way to give the original signal levels.

Since this active crossover is adjustable, you can set the crossover frequency to be at the ideal point for your loudspeaker. You can also adjust the steepness of the roll-off, as shown by the dotted lines, as different roll-off rates suit different situations.

There's also an optional subsonic filter, so that very low (inaudible) frequencies, or those which are too low for the woofer to reproduce, are eliminated and do not waste your amplifier power or possibly damage your woofer. Its frequency is also adjustable. (This is essential for vented, horn-loaded and infinite-baffle speakers).

The relative levels of the woofer and tweeter can also be adjusted, to compensate for differing driver efficiencies or amplifier gains, and although it isn't shown on the diagram,

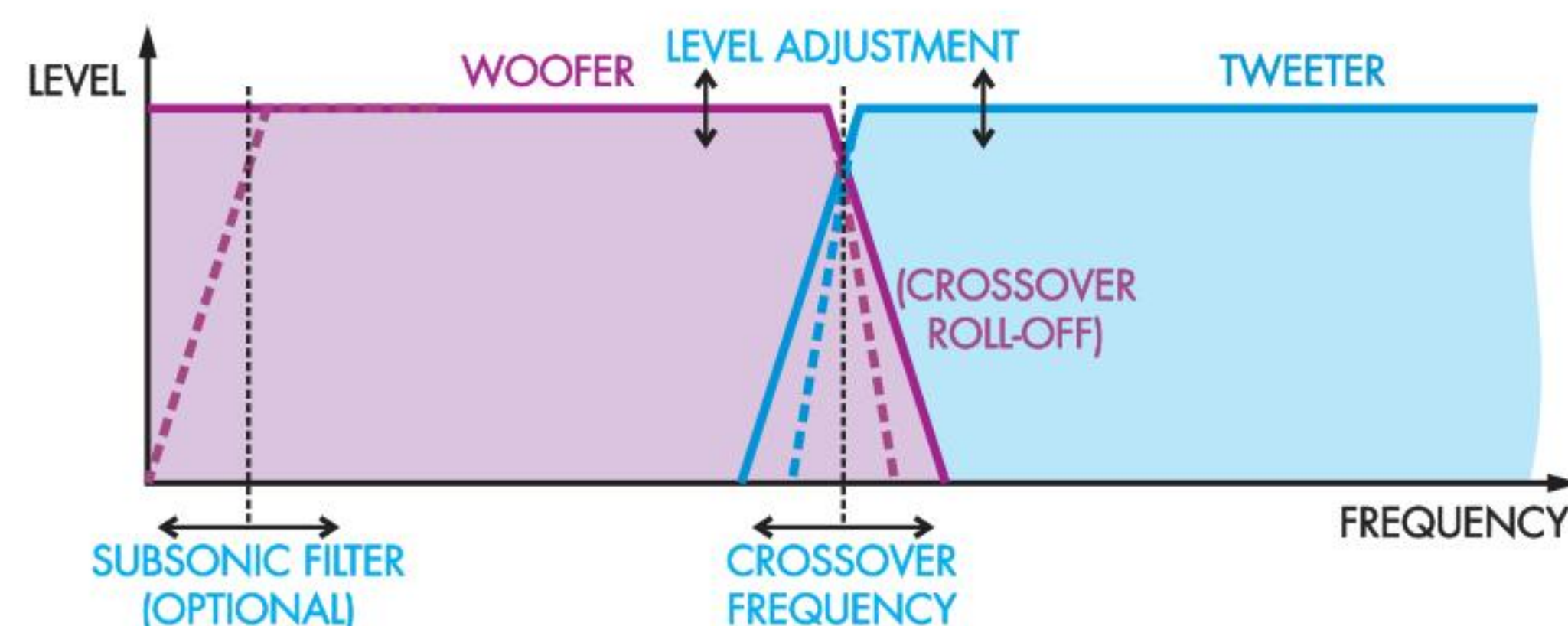


Fig.1: this two-way Active Crossover splits a signal with a spectrum covering the entire audible frequency range into two signals, one with the components above the crossover frequency and the other with the components below it. The optional woofer high-pass filter removes subsonic signals.

you can also delay one channel slightly relative to the other, to give proper 'time alignment'.

The four parametric equalisation channels are not shown in Fig.1, but essentially, each can be configured as either a high-pass or low-pass filter with adjustable stopband attenuation and corner frequency. This allows you to 'shelve' frequencies above or below a specific frequency, or between or outside a pair of frequencies, to shape the overall frequency response at all four outputs.

The *Active Crossover* is used as shown in Fig.2. It's connected between the stereo outputs of a preamp and four power amplifiers which power the four loudspeaker drivers independently.

Note that you don't need to use a preamplifier because this *Active Crossover* has a built-in volume control, so you can use it as a basic preamp too. In that case, the signal source is connected directly to the *Active Crossover's* inputs.

Why use an active crossover?

There are a few reasons why you may want to use an active crossover. First, if you are building speakers from scratch, it's probably easier to use an active crossover than design a passive one, since you can easily experiment with it and change the crossover frequency/frequencies, relative amplitudes and so on until it sounds 'right'.

Also, if you're building a seriously powerful system with big amplifiers and big speakers, it's difficult (and expensive) to design a passive crossover to handle all that power.

Since an active crossover is connected before the amplifiers, and the amplifiers can then power the drivers with nothing in between, efficiency is maximised and you can deliver as much power as your amplifiers and drivers can handle.

Depending on the speaker design, you may also wind up with better overall sound quality using an active crossover than a passive one. Partly this is because it's hard to create a very 'steep' passive crossover, which crosses over across a small frequency range, but this is relatively easy to do with an active crossover.

Also, when using an active crossover, especially a digital one, because you have separate line-level signals for the tweeters and woofers, it is possible to compensate for the slightly different distance from each diaphragm to the listener by delaying one of the signals.

The exact delay required depends on the driver and cabinet design; it's tough to achieve perfect 'time alignment' mechanically, so being able to adjust this electronically is a boon.

Another advantage of an active crossover is that if you drive the system into clipping, usually this will be due to a

Features and specifications

- Low distortion and noise: ~0.001% THD+N
- One stereo input, two stereo outputs (low/high), optional channel inversion
- Each pair of outputs can be crossed over using first, second or fourth-order digital filters
- Additional parametric equalisers: four, common to all outputs
- Optional high-pass filter for low-frequency outputs, to cut out subsonic frequencies
- Configurable delay for each channel, to compensate for driver offsets (up to 6.2m; 18ms)
- Individually configurable output inversion and attenuation settings
- Built-in volume control – no need to use a preamp
- Load and save setups to EEPROM
- Software written in Microchip C; could be adapted for other DSP uses (open source).

huge bass signal. With a single amplifier for each of the left and right channels, that means that the treble signal will be clipped off entirely each time the bass signal hits one of the rails. That can sound really bad.

But with bi-amplification, even if you're clipping the bass signal, since most of the treble is going through a separate amplifier, it won't be affected. The result will still not be ideal, but won't sound anywhere near as bad; be thankful for small mercies!

Basically, except for the extra complexity that comes with the use of an active crossover, there are only benefits to this arrangement. It's much easier to adjust and tweak to give near-ideal sound quality, has minimal effect on signal quality or speaker power handling and can be adapted to any two-way loudspeaker system, as long as you can wire up each driver separately.

Modular design

This *DSP Crossover* is built by combining several different modules, each with a specific function. It was designed this way so that it could be reconfigured to do many different audio DSP tasks. In fact, with the same hardware but different software, it could be used for a variety of audio processing tasks such as echo/reverb/effects, equalisation, delay and so on.

The basic configuration is shown in Fig.3. It uses seven main boards: one stereo analogue-to-digital converter (ADC) board, two stereo digital-to-analogue converter (DAC) boards, a microprocessor board, a power supply/signal routing board and a front panel interface board. These are rounded out with a graphical LCD module for display, and a mains transformer to power it.

Interconnections are made between the boards with ribbon cables fitted with standard insulation displacement (IDC) connectors. This is a convenient and easy way to join boards where multiple signals and power need to be routed between them.

Audio signals are fed into the unit via the ADC board, where they are converted to digital data. This data passes through the

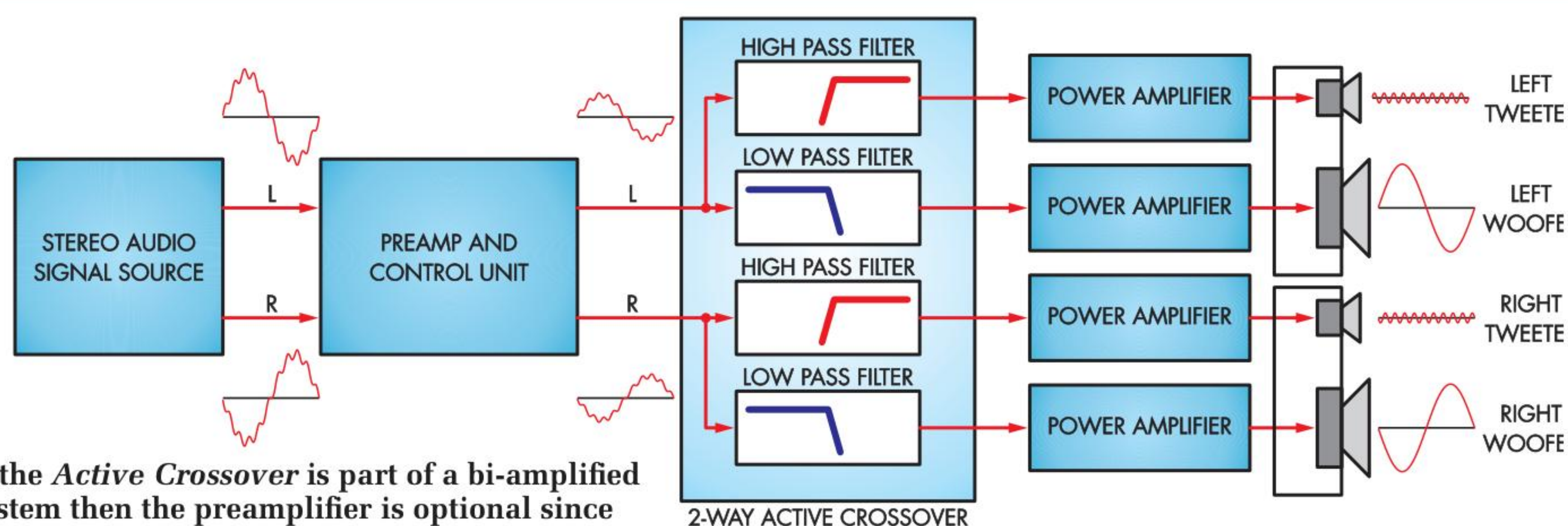
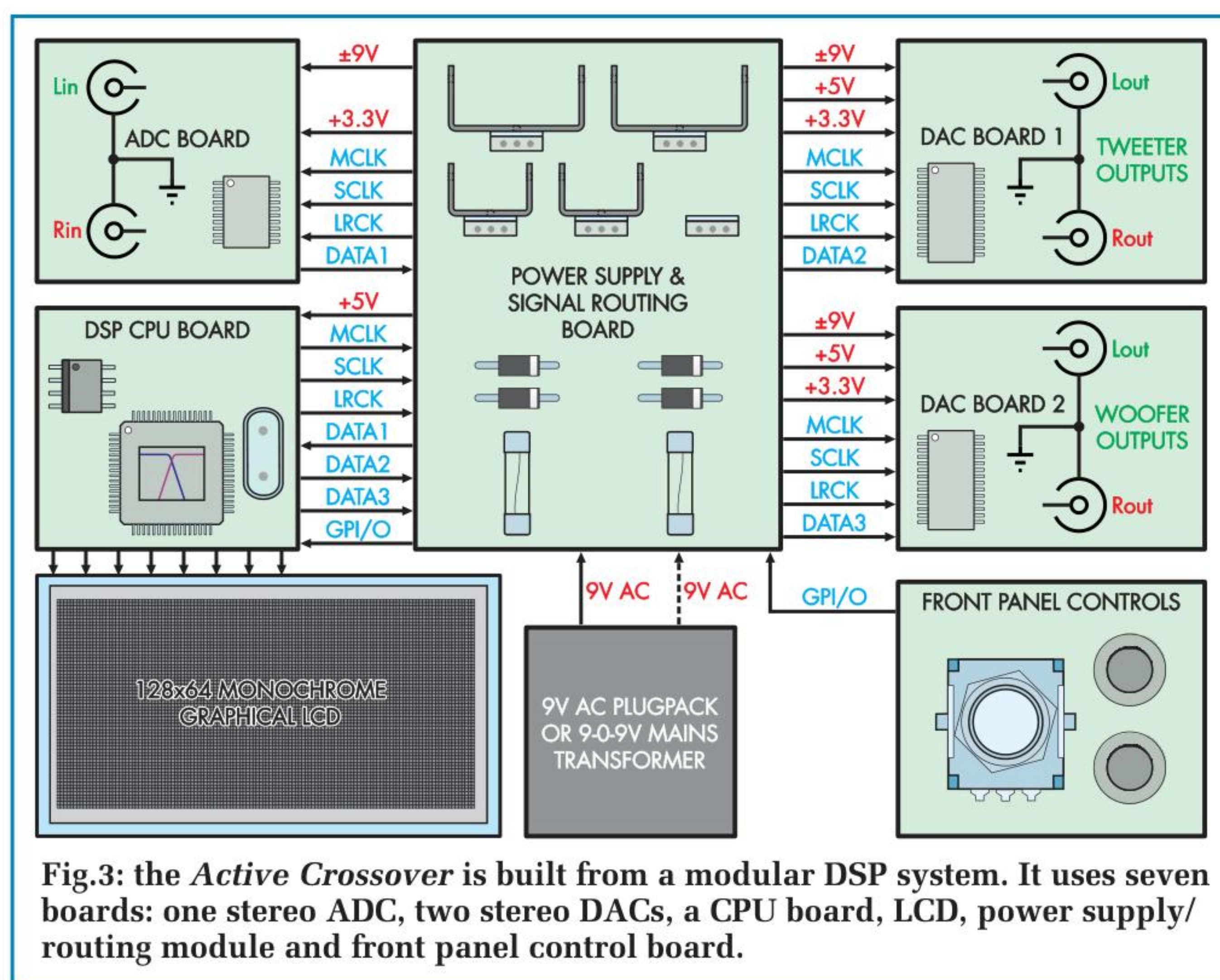


Fig.2: if the *Active Crossover* is part of a bi-amplified Hi-Fi system then the preamplifier is optional since the *Active Crossover* has built-in volume control.



power supply/routing board and onto the microcontroller, which stores it in RAM before doing whatever processing is necessary.

It then feeds this data back out through a different set of pins, again as serial digital audio data, where it passes back through the routing board and onto one (or both) of the DAC modules. The DAC modules then convert these digital signals back into line-level analogue signals which are available from two RCA connectors on the rear panel.

The microcontroller board is wired directly to the graphical LCD, so it can show the current status and provide the user interface, while the separate front panel control board connects to the micro via the routing board, allowing the user control over that interface.

The whole thing is powered from a 9V transformer, which could be a plugpack or mains type. If a mains transformer is used, it would generally be an 18V centre-tapped (9-0-9V) type, to give full-wave rectification.

However, half-wave rectification, as is the case with most plugpacks (they usually have a single secondary winding), is good enough.

Circuit description

Let's start where the audio signals enter the unit, the ADC board. The circuit diagram for this board is shown in Fig.4. It's built around an ultra high-performance ADC, the CS5361 (IC1), which has a dynamic range of 111dB and a typical THD+N figure of 0.001%. (There is a compatible alternative, the CS5381, which offers even lower distortion.)

The stereo line-level audio signals are fed in via RCA sockets CON1a and CON1b. They pass through ferrite beads

with 100pF capacitors to ground, both intended to remove any RF signals, either from the signal source or picked up in the connecting leads. As the two channels are processed identically before they reach the inputs of IC1, we'll just describe the left channel path.

The audio signal is then AC-coupled to non-inverting input pin 3 of op amp IC2a, an NE5532 low-noise, low-distortion device. Schottky diodes D1 and D2 prevent excessive voltages from being applied to this op amp; eg, inductive spikes generated by lightning or from incorrectly connected equipment. A 100kΩ resistor to ground provides a path for direct current to flow out of that input pin.

IC2a buffers the signal, providing a low-impedance source for the following filters. The signal is then fed to op amp IC2b; an inverting amplifier with a gain of -1, due to the use of two resistors of the same value in the feedback network.

A 33pF capacitor across the resistor between pins 7 (output) and 6 (inverting input) rolls off the ultrasonic frequency response to provide stability.

The reason for this inverting stage is that the ADC chip (IC1) is a differential design, so for both the left and right channel inputs, it expects two signals, one 180° out of phase with the other. The in-phase signal comes from the output (pin 7) of IC2b, while the out-of-phase signal is taken directly from the output (pin 1) of the preceding buffer, IC2a. It may seem odd that the in-phase signal comes from the output of the inverter, but this is because the following filter stages are also inverting, so it will end up with the same phase as the inputs, while the other signal will be out of phase.

Both signals are then fed through identical buffer/filter arrangements, built around IC4a and IC4b. These filters are similar to what is recommended in the CS5361 data sheet (see Figure 24 at: <http://bit.ly/pe-jan20-adc>), but not identical. The data sheet says: 'The digital filter will reject signals within the stopband of the filter. However, there is no rejection for input signals which are (n×6.144 MHz) the digital passband frequency, where n=0,1,2, ... Refer to Figure 24 which shows the suggested filter that will attenuate any noise energy at 6.144 MHz, in addition to providing the optimum source impedance for the modulators.'

The main difference between our circuit and the recommended circuit is that ours is inverting. While inverting amplifiers introduce more noise than non-inverting amplifiers, inverting amplifiers can have lower distortion due to their near-zero common-mode voltage. Also, the use of inverting amplifiers allows us to easily provide a slightly different DC bias to the two signals.

This is done by connecting a low-value resistor (8.2Ω) between the non-inverting input pins (pins 3 and 5) of op amps IC4a/IC4b, which are in series with a divider across the supply rail (10kΩ/10kΩ).

The reason for DC biasing the two differential inputs differently is to overcome a potential problem with analogue-to-digital converters – when the signal is near the 'zero point' the binary values at the output tend to flip between all zeros and all ones. This can cause digital noise at the worst possible time – when there is near silence at the inputs.

By adding a slight DC offset, the zero point is moved such that any small amount of noise will only cause a few bits to flip. That offset is removed by digital filtering inside the ADC chip. While modern delta-sigma ADCs do not suffer from this problem anywhere near as severely as early ADCs, this solution is cheap insurance to guarantee that the bit-flipping problem does not affect us.

The bottom end of the divider which produces the half-supply bias rails is bypassed with 10μF and 100nF capacitors, to reject any noise and ripple that may be on this rail, and prevent it from getting into the signal path. The ADC runs from its own regulated 5V rail which should be pretty 'quiet'. But this is a very high-performance ADC, so it isn't worth taking any risks in feeding noise into its inputs.

The 91Ω series resistors at the op amp outputs protect the ADC from excessive voltages. The op amps run from ±9V while the ADC runs from 5V, so the op amps outputs can swing beyond both of

STORE YOUR BACK ISSUES ON CD-ROMS

- VOL 1 BACK ISSUES** – January 1999 to June 1999
Plus bonus material from Nov and Dec 1998
- VOL 2 BACK ISSUES** – July 1999 to December 1999
- VOL 3 BACK ISSUES** – January 2000 to June 2000
- VOL 4 BACK ISSUES** – July 2000 to December 2000
- VOL 5 BACK ISSUES** – January 2001 to June 2001
- VOL 6 BACK ISSUES** – July 2001 to December 2001
- VOL 7 BACK ISSUES** – January 2002 to June 2002
- VOL 8 BACK ISSUES** – July 2002 to December 2002
- VOL 9 BACK ISSUES** – January 2003 to June 2003
- VOL 10 BACK ISSUES** – July 2003 to December 2003
- VOL 11 BACK ISSUES** – January 2004 to June 2004
- VOL 12 BACK ISSUES** – July 2004 to December 2004
- VOL 13 BACK ISSUES** – January 2005 to June 2005
- VOL 14 BACK ISSUES** – July 2005 to December 2005
- VOL 15 BACK ISSUES** – January 2006 to June 2006
- VOL 16 BACK ISSUES** – July 2006 to December 2006
- VOL 17 BACK ISSUES** – January 2007 to June 2007
- VOL 18 BACK ISSUES** – July 2007 to December 2007
- VOL 19 BACK ISSUES** – January 2008 to June 2008
- VOL 20 BACK ISSUES** – July 2008 to December 2008
- VOL 21 BACK ISSUES** – January 2009 to June 2009
- VOL 22 BACK ISSUES** – July 2009 to December 2009
- VOL 23 BACK ISSUES** – January 2010 to June 2010

FIVE-YEAR DVD/CD-ROMs

- No.1** – Jan 03 to Dec 07
- No.2** – Jan 04 to Dec 08
- No.3** – Jan 05 to Dec 09
- No.4** – Jan 06 to Dec 10
- No.5** – Jan 07 to Dec 11
- No.6** – Jan 08 to Dec 12
- No.7** – Jan 09 to Dec 13
- No.8** – Jan 10 to Dec 14
- No.9** – Jan 11 to Dec 15
- No.10** – Jan 12 to Dec 16
- No.11** – Jan 13 to Dec 17
- No.12** – Jan 14 to Dec 18 – **NEW!**

ONLY £16.45 each
Five years just £29.95
including VAT and p&p

- VOL 24 BACK ISSUES** – July 2010 to December 2010
- VOL 25 BACK ISSUES** – January 2011 to June 2011
- VOL 26 BACK ISSUES** – July 2011 to December 2011
- VOL 27 BACK ISSUES** – January 2012 to June 2012
- VOL 28 BACK ISSUES** – July 2012 to December 2012
- VOL 29 BACK ISSUES** – January 2013 to June 2013
- VOL 30 BACK ISSUES** – July 2013 to December 2013
- VOL 31 BACK ISSUES** – January 2014 to June 2014
- VOL 32 BACK ISSUES** – July 2014 to December 2014
- VOL 33 BACK ISSUES** – January 2015 to June 2015
- VOL 34 BACK ISSUES** – July 2015 to December 2015
- VOL 35 BACK ISSUES** – January 2016 to June 2016
- VOL 36 BACK ISSUES** – July 2016 to December 2016
- VOL 37 BACK ISSUES** – January 2017 to June 2017
- VOL 38 BACK ISSUES** – July 2017 to December 2017
- VOL 39 BACK ISSUES** – January 2018 to June 2018
- VOL 40 BACK ISSUES** – July 2018 to December 2018
- VOL 41 BACK ISSUES** – January 2019 to June 2019

15 YEARS OF EPE!

AMAZING BUNDLE OFFER

- No.1 – Jan 03 to Dec 07**
PLUS
No.6 – Jan 08 to Dec 12
PLUS
No.11 – Jan 13 to Dec 17

ALL FOR JUST
£39.95

All DVD/CD-ROMs are suitable for any PC with a DVD/CD-ROM drive.
They require Adobe Acrobat Reader (free from: www.adobe.com/acrobat)

ORDER YOUR CD-ROMs TODAY!
JUST CALL 01202 880299 – OR VISIT www.epemag.com

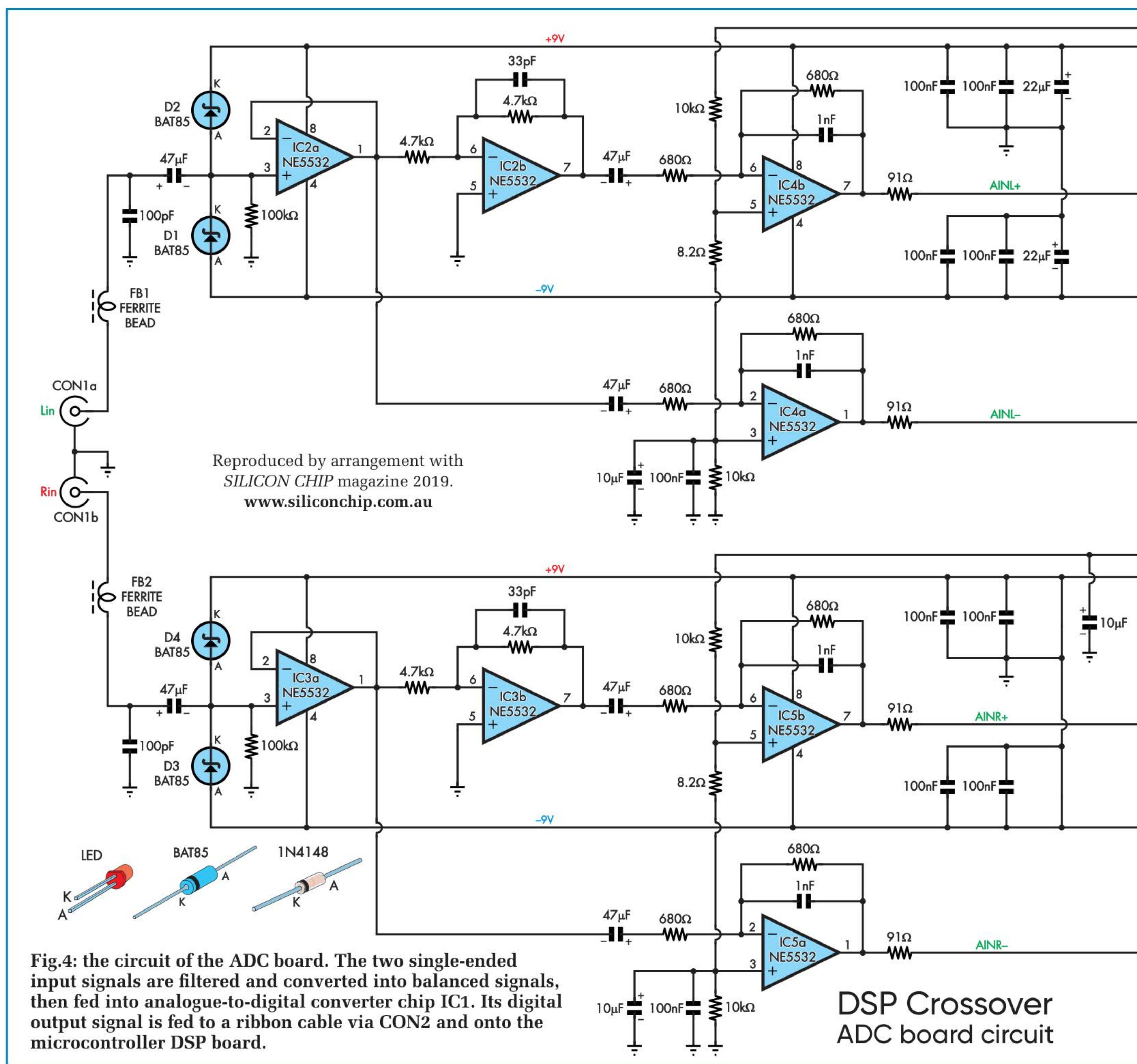


Fig.4: the circuit of the ADC board. The two single-ended input signals are filtered and converted into balanced signals, then fed into analogue-to-digital converter chip IC1. Its digital output signal is fed to a ribbon cable via CON2 and onto the microcontroller DSP board.

the ADC supply rails. However, since the op amp feedback comes from after this resistor (ie, it's inside their feedback loops), the output impedance is still very low, and the frequency response is flat.

Schottky diodes D5, D6, D9 and D10 help to further protect the ADC inputs, by conducting if the op amps try to drive the ADC inputs below -0.3V or above $+5.3\text{V}$. This prevents any standard silicon devices (eg, transistors or diodes) inside IC1 from conducting due to an excessive input voltage, as usually this will only happen once the applied voltage is more than 0.6V beyond the supply rails.

The 91Ω resistors also combine with a 2.7nF capacitor across the differential inputs of IC1, to provide some further differential filtering, to keep out any signals at 6.144MHz or above (the ADC's internal clock rate), which could affect the signal quality through aliasing.

Analogue-to-digital conversion

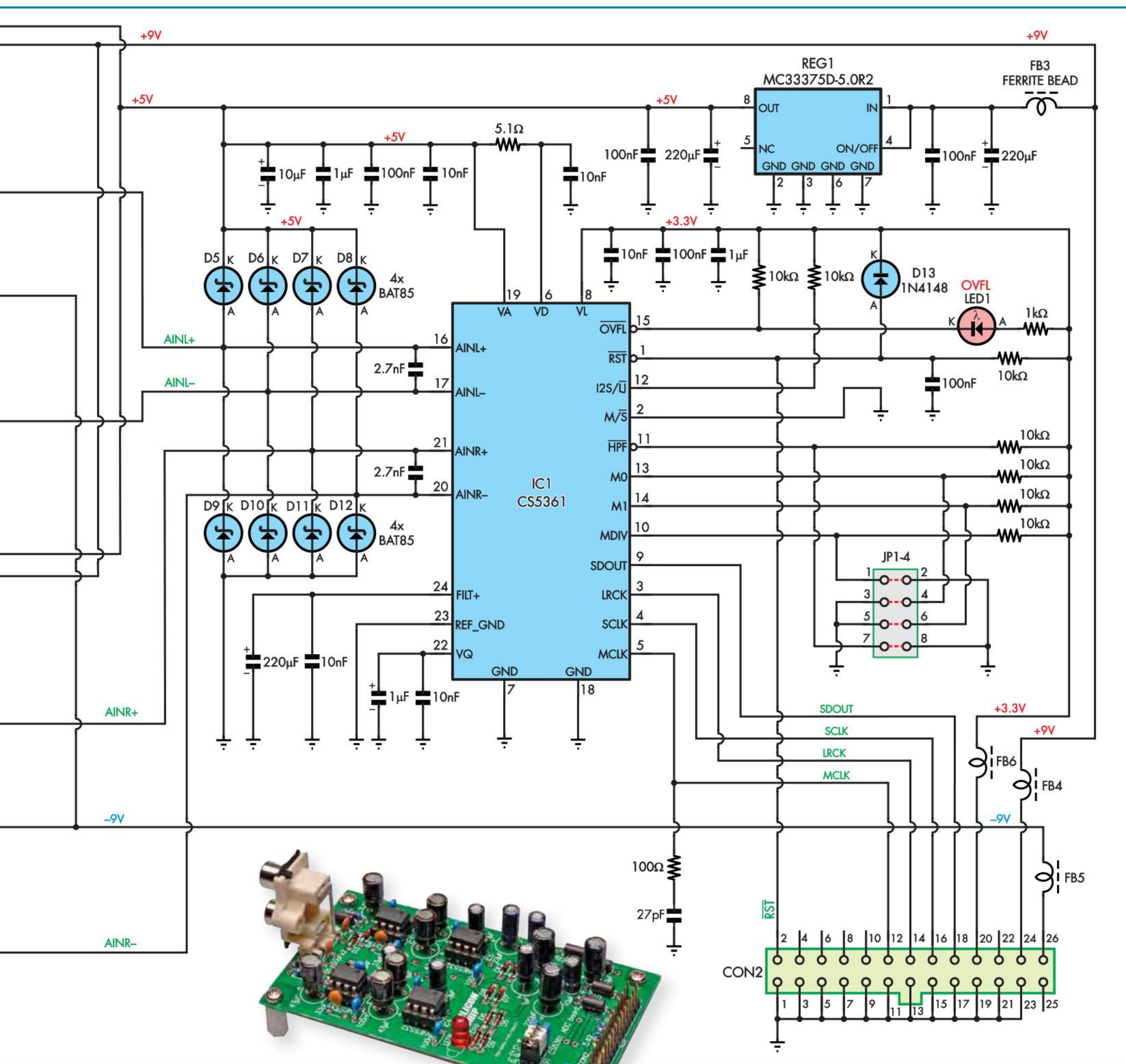
The stereo differential signals are applied to input pins 16, 17, 20 and 21 of IC1. There are some extra components connected to this IC, which are required for its correct operation.

It has two internal reference voltages, which are fed to pins 22 (VQ or quiescent voltage) and 24 (FILT+) and these need to be externally bypassed to ground via capacitors. We have provided two capacitors to filter each of these rails, 10nF in both cases, plus $220\mu\text{F}$ for FILT+ and $1\mu\text{F}$ for VQ. The use of two different values provides a lower impedance across a broader range of frequencies.

IC1 has three different supply pins: VA (pin 19) for the analogue 5V supply, VD (pin 6) for the digital 5V supply and VL (pin 8) for the 3.3V logic/interface supply. The supply arrangement is described below.

Pin 1 is IC1's reset input, and this is connected to the logic supply via a diode and resistor, and to ground via a capacitor. This forms a power-on reset circuit. Initially, the capacitor is discharged and so the reset input is low, resetting IC1. This capacitor then charges up via the $10\text{k}\Omega$ resistor and releases reset after a few milliseconds. When power is switched off, the capacitor rapidly discharges via D13.

This reset pin is also connected to pin 2 of CON2, which is routed to the microcontroller, so it can reset IC1 after power-up if necessary.



Pin 2 selects either master mode (when high, ie, IC1 drives the digital audio clock lines) or slave mode (when low, ie, IC1 is clocked externally). This is connected directly to ground since the audio clock signals are supplied from the microcontroller via pins 12, 14 and 16 of CON2. These connect to pins 5, 3 and 4 of IC1 respectively, and in slave mode, these are the clock inputs.

Pin 5 (MCLK) is the master (oversampling) clock, which is typically around 12.288MHz, ie, $48\text{kHz} \times 256$. This is used to clock the ADC modulator and other internal circuitry. Pin 3 is the left/right clock or sample clock, and this is usually at around 48kHz. When it is high, the serial data pin is normally carrying left audio channel data; when it is low, right audio channel data.

Pin 4 is the sample clock and this clocks the serial data itself. It usually operates at the sampling rate multiplied by the number of channels, eg, $48\text{kHz} \times 2 = 96\text{kHz}$. The serial data comes from pin 9 of IC1 and goes to pin 18 of CON2, where it eventually feeds into the microcontroller.

Note that pin 5 (MCLK) of IC1 has a snubber network connected to ground. This is intended to prevent ringing and is a

good idea when a high-frequency signal is fed through a long wire; however, at 12.288MHz it was found not to be necessary, and so those components can be safely left off.

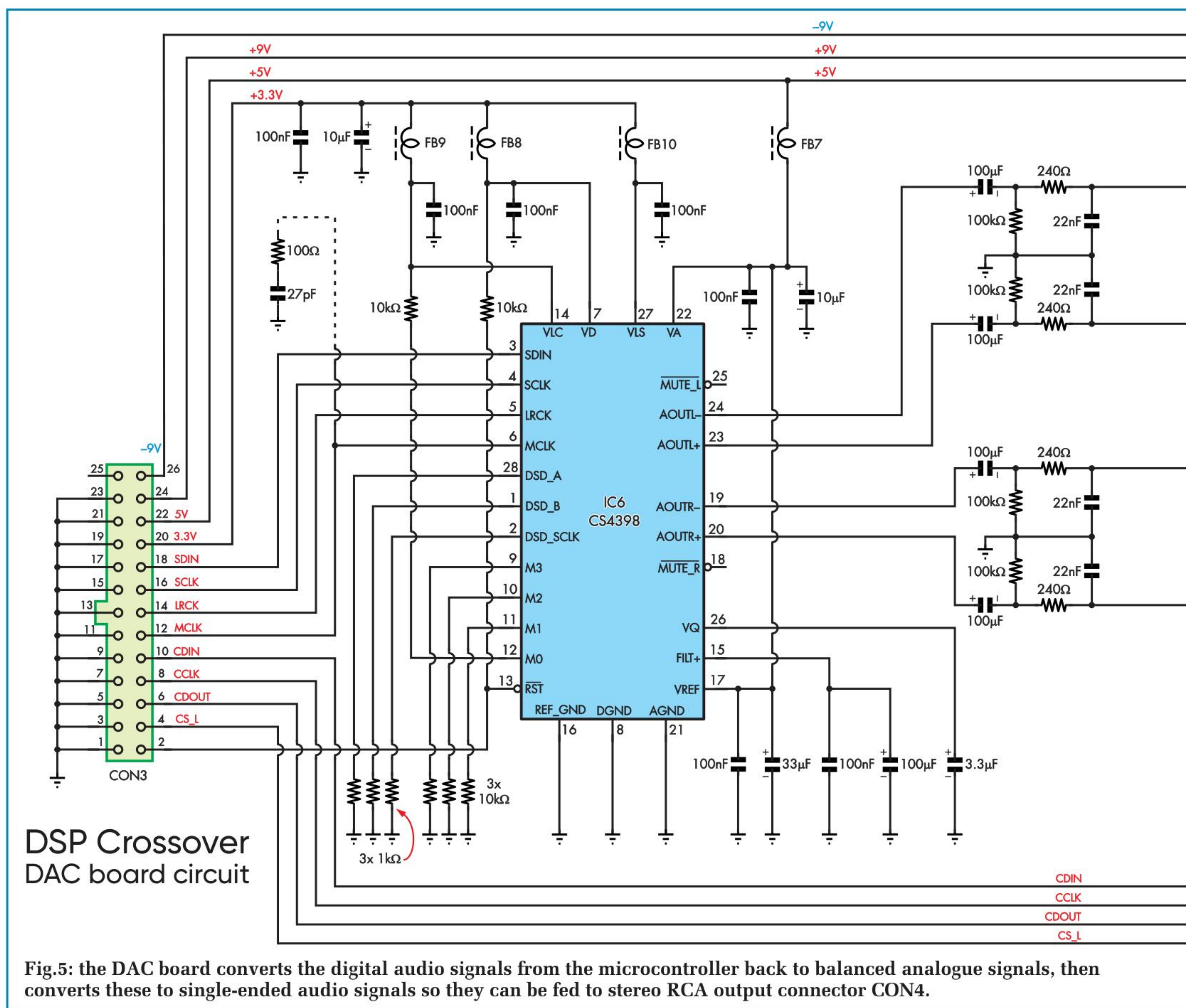
ADC configuration

Pins 10-14 of IC1 are configuration inputs, and their state determines how the ADC operates.

Pin 10 (MDIV) causes the master clock signal to be divided by two when high, allowing a higher frequency master clock to be used. Pin 11 enables or disables a digital high-pass filter, to remove any DC offset from the input signals. Pin 12 selects the digital audio output data format, either I²S or left-justified.

Pins 13 and 14 select the sampling rate range, either single-speed mode (2-51kHz, M0 and M1 low), double-speed mode (50-102kHz, M0 high) or quad-speed mode (100-204kHz, M1 high).

Of these five pins, pin 12 (I²S/L^J) is tied to VL via a 10kΩ resistor, permanently selecting I²S format. The other four connect to jumpers JP1-JP4 and have 10kΩ pull-ups to VL. So they are high by default but can be pulled low by placing a shorting block on the jumper. Typically, all four of the jumpers are fitted, so that master clock division is



disabled, the high-pass filter is enabled and the sampling rate can be 48kHz. However, note that the use of jumpers means you could change the software (eg, to use a higher sampling rate) and easily reconfigure the ADC board to suit.

Pin 15 of IC1 goes low if either input signal swings outside the range that the ADC can cope with. We have an LED (LED1) connected to this pin, with a 1k Ω current-limiting resistor to VL. So LED1 will light if the input signal level is too high for IC1 to cope with, resulting in digital clipping.

Power supply rails

The 5V analogue supply comes from the output of an MC33375D low-dropout regulator, REG1, which is fed from the incoming +9V supply via a ferrite bead (FB3). This regulator was chosen for its very tight line and load output specifications (2mV and 5mV respectively), which means that the resulting analogue 5V rail should be very stable indeed.

REG1 has 100nF and 220 μ F input bypass and output filter capacitors, but there are also four bypass capacitors right near IC1's VA input pin: 10nF, 100nF, 1 μ F and 10 μ F. Again, these different values were paralleled to provide a very low supply source impedance for IC1 across a wide range of frequencies, from a few hertz up to many megahertz.

The 5V digital supply (VD) is powered from the same 5V rail as VA, but with a 5.1 Ω resistor in between so that

digital noise does not feed back into the analogue supply. The VD rail has a separate 10nF bypass capacitors for high-frequency stability.

The 3.3V logic supply comes from pin 20 of interface header CON2, via another ferrite bead (FB6) and is bypassed with 10nF, 100nF and 1 μ F capacitors.

The $\pm$ 9V supply rails for the op amps (also used to derive the 5V rails) are fed in via pins 24 and 26 of box header CON2, with series ferrite beads to stop RF signals from propagating in either direction. This is important because long unshielded ribbon cables can pick up all sorts of EMI.

Microcontroller interface

CON2 carries the power supply, control and digital audio signals. It's a 26-pin DIL header which connects to a ribbon cable. By tying all odd numbered pins to ground (except for pin 25), every second wire in the ribbon cable is grounded, minimising interference between adjacent signals on the even-numbered pins.

As discussed, pins 20, 24 and 26 power the ADC board, while pins 12, 14, 16 and 18 carry the clock signals and digital audio data, and pin 2 is the reset line. Pins 22 and 25 are unused; pins 4, 6, 8 and 10 are reserved for an SPI control bus. In fact, IC1 does not have an SPI control interface, so those pins are not routed anywhere on this board.

filtering, based around either IC9a or IC9b. This also converts them to single-ended signals.

These stages provide some gain, to boost the ~1V RMS from the DAC up to around 2.3V RMS, a similar level to that produced from many other audio sources; for example, CD/DVD/Blu-ray players

The signals are then AC-coupled by 22µF capacitors and DC-biased to ground using 10kΩ resistors, to remove any remaining DC bias on the signals. They are then fed to the inputs of IC10, a PGA2320 volume-control chip.

There are two things to note about this chip. One is that we're feeding the left channel signal to its right channel input and the right channel signal to its left channel input. But that doesn't matter since its channels are independent.

The other is that the CS4398 already has a built-in digital volume control. IC10 is included on the board because it adds little noise to the signal and since the signal swing is higher at the outputs, we thought that this would introduce less distortion. And that is true, but the effect is quite small, so we didn't even bother wiring up the control signals from IC10 to the microcontroller.

So you can leave it off the board and instead, solder 0Ω resistors from its pin 9 pad to pin 11, and another from pin 16 to pin 14, so that the signals from IC9 go straight to the output RCA connectors, CON4.

While it may seem odd that there's a footprint for IC10 when it isn't connected to the microcontroller, it could be useful if the board was used in a different project, and there was space on the board, so we've left the option open.

Power supplies

Like the ADC board, the op amps run off the ±9V supplies fed in from the power supply board via CON3. However, rather than passing through ferrite beads, on this board each op amp has a 10Ω/100µF RC low-pass filter for each supply rail, as well as 100nF bypass capacitors for each op amp supply pin.

Another difference from the ADC board is while that board derived a local 5V supply from +9V using an onboard regulator, on this board, DAC IC6 and (if fitted) volume control IC10 run from a 5V supply that's fed from the power supply board, via pin 22 of CON3.

The two chips have separate ferrite beads on this supply line for isolation, plus small and large bypass capacitors.

DAC IC6 also requires three 3.3V supply rails – one for I/O (VLC, pin 14), one for its digital circuitry (VD, pin 7) and one for its internal PLL (VLS, pin 27).

These are all powered from the same 3.3V supply rail via pin 20 of CON3, but again they have separate ferrite beads for EMI suppression and isolation, plus individual 100nF bypass capacitors.

There are also 100nF and 10µF capacitors on the incoming 3.3V supply rail.

Volume control

As mentioned earlier, volume control chip IC10 is not required, but if it is fitted, it is powered from the ±9V rails (at the VA+ and VA- pins) and also from the 5V rail via ferrite bead FB11. The ZCEN input (pin 1) is pulled up to +5V with a 10kΩ resistor, while Mute (pin 8) is similarly pulled up by a 10kΩ resistor.

Pin 1 is the Zero Crossing Enable control, and when pulled high it will wait for the audio signal to cross through 0V before making any volume changes. This avoids clicks which would otherwise be caused by a sudden signal level step change when the volume is adjusted.

Unsurprisingly, pulling pin 8 low mutes the output, and this function is not used, hence the pull-up resistor. Mute can instead be controlled using the SPI serial control interface.

Power supply and signal routing board

Let's turn now to the power supply and signal routing circuits, shown in Fig.6. The cable from CON1 on the ADC board connects to CON16, while two separate but identical DAC boards are connected to CON14 and CON15.

10-way headers CON17 and CON18 connect to the microcontroller board. The signals to and from the ADC and DAC boards are routed to the microcontroller pins via these headers. At the same time, five power rails are distributed to all those boards as required.

Except for the master clock, all the signals from CON18 are connected through to CON19, which the front panel control board plugs into. This routes the control board signals back to the microcontroller.

Here are some things to note about the signals passing between the micro and ADC/DAC boards: CON14 (DAC1) and CON16 (ADC) share the same digital audio bus, while CON15 has a separate bus. One DAC and one ADC module can share the same bus since there is one pair of data in/out lines and they only use one each (into the DAC, out from the ADC).

The same master clock signal is distributed to all three connectors, and the reset line is also shared between all three, so the three chips will be reset simultaneously if this line is pulled low.

None of the SPI control buses are wired up to anything, as this is not required as long as you leave the volume control chips off the DAC boards.

The ADC and DAC boards are fed with +9V, -9V, +5V (VA, not used by the ADC board) and +3.3V (to power the digital interfaces of the ADC and DACs). A separate 5V rail passes through ferrite bead FB15 and is then fed to the microcontroller board, to power the micro. Using a separate rail avoids the possibility of the micro board 'polluting' the 5V rail used by the DAC boards.

All the digital audio signals connect to the micro via CON17 (along with its 5V supply), except for the master clock, which is on pin 8 of CON18. The other pins on CON18 are wired to general purpose I/Os on the microcontroller.

The power supply section is pretty straightforward: a centre-tapped 18-24V AC (eg, 12 + 12V AC) transformer is wired to CON13 and then connects to diode bridge rectifier D17-D20 via fuses F1 and F2.

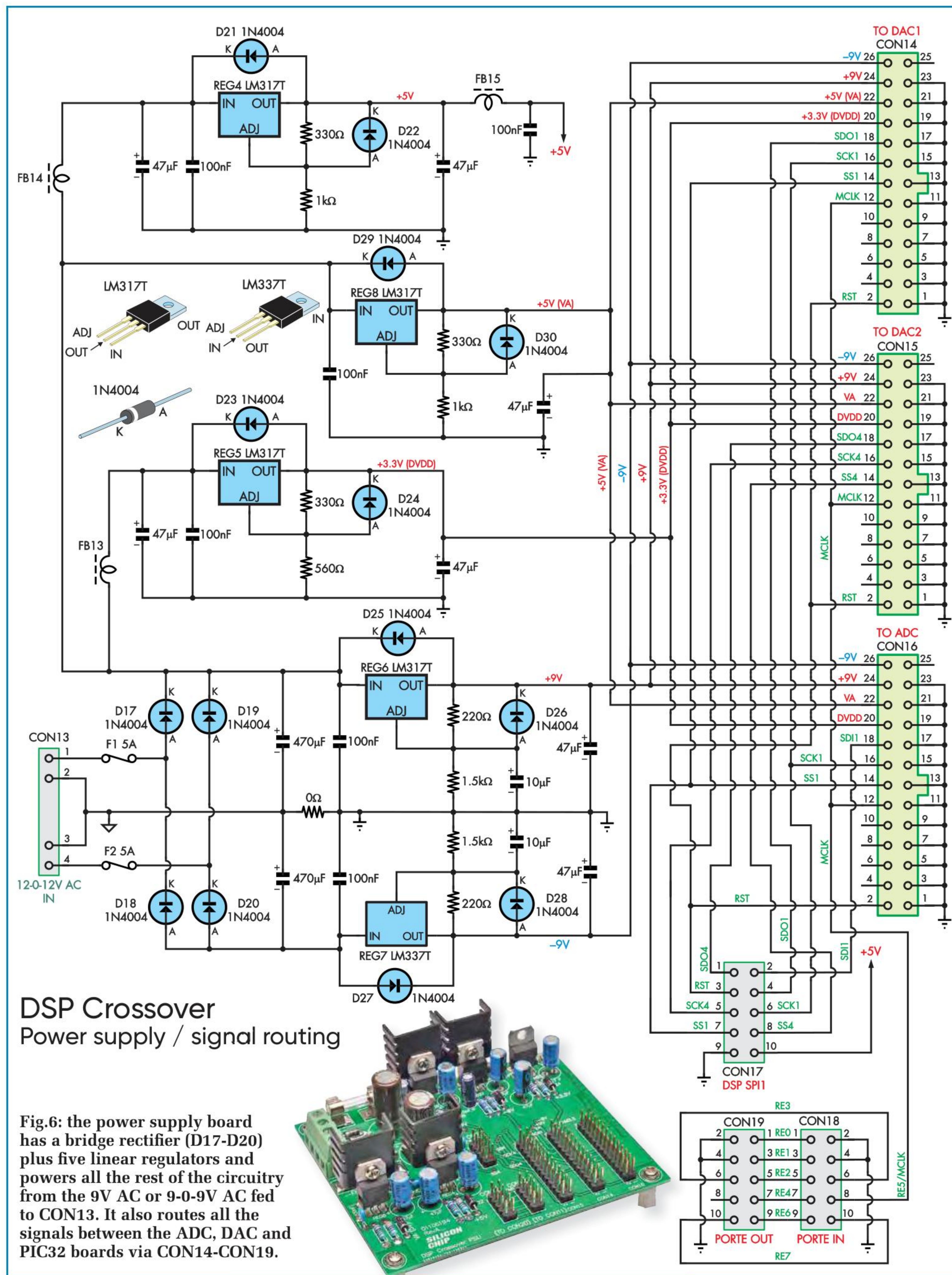
The DC outputs of this bridge are filtered by a pair of 470µF capacitors and then regulated by adjustable regulators REG6 and REG7 to produce the +9V and -9V rails respectively.

LM317/337 adjustable regulators are used because of their excellent ripple rejection capability, especially with 10µF capacitors from their ADJ terminals to ground. The 220Ω and 1.5kΩ resistors set their nominal output voltages to $(1.5k\Omega/220\Omega + 1) \times 1.2V = 9.38V$. The extra diodes protect the regulators by preventing current from flowing backwards through them at switch-off. These regulators are fitted with small flag heatsinks to keep their temperatures reasonable.

The positive output of the bridge rectifier is also fed through ferrite beads FB13 and FB14 through to two extra



The project uses a 128 × 64 graphical LCD for set up and to view the configuration. It is controlled using a rotary encoder and two pushbuttons to drive the menu-based interface.



47µF capacitors which power regulators REG4 and REG5 respectively, to produce the +5V and +3.3V rails. Different feedback resistor values are used to change the LM317 output voltages. The extra ripple-rejection capacitors are not

used here since these supplies do not need to be as 'quiet'. Another LM317, REG8, is fed from the main 470µF positive filter capacitor and is also set up for a 5V output. This provides the 5V 'VA' rail for both DAC boards.

Coming up

We can't fit all of the project's details into one article. In the next two issues, we'll cover the microcontroller and front panel circuits, along with the parts, construction and operation.

ISOLATED SERIAL LINK

by Tim Blythman

Want to communicate with and/or program a micro that's connected to mains or a high-voltage supply? Hmmmm . . . r-i-s-k-y . . . not just to the device, but to you as well! Here's the *safe* way to do it.

Because small computer boards like the Micromite, Arduino and Raspberry Pi are so flexible, chances are you will eventually find yourself using them to control some mains-powered or high-voltage battery-powered circuitry.

But there's always the risk that those higher voltages could find their way back to your computer, doing untold damage – and in the worst case, it could be *you* that suffers the untold damage!

This nifty little project allows you to send serial data over an optically isolated link, entirely preventing the dreaded 230V-in-the-USB-socket-instant-death syndrome.

It can be used for programming the project you are working on, or for monitoring and feedback from a finished project to your computer.

Either way, it provides total isolation. It can also translate 5V serial signals to 3.3V, and vice versa. You can even use it to pass data between the USB ports on two separate computers without having to make an electrical connection between the two, avoiding the possibility of earth loops or other similar problems.

You may have seen our *USB Port Protector* project in the May 2019 issue. If so, you'll understand our motivation for this project (sob!). But this provides even better protection for your PC. It doesn't try to shunt excessive voltages and currents – it won't even let them near your computer!

We've had laptop USB ports fail while plugged into certain Arduino work-in-progress projects which involved mains and battery power. We aren't exactly sure how it happened, but it appears that some voltages got to certain pins that they were not supposed to. We wish we'd had this *Isolated Serial Link* then; it's an expensive lesson to learn!

Projects that feature high voltages and high currents always have the potential for damage to delicate components like microcontrollers and even computers. Where possible, it is best to separate the two. This circuit is simple, easy to build and does just that. It's also useful for situations even where there are no USB ports involved, eg, to allow two microcontrollers to communicate via a serial link, even if they are running from different supplies which may not share a common ground.

Shown here significantly over size for clarity (actual board size is 74mm square), this version has a transformer-based power supply (in case the target PCB doesn't have its own supply.)

What does it do, exactly?

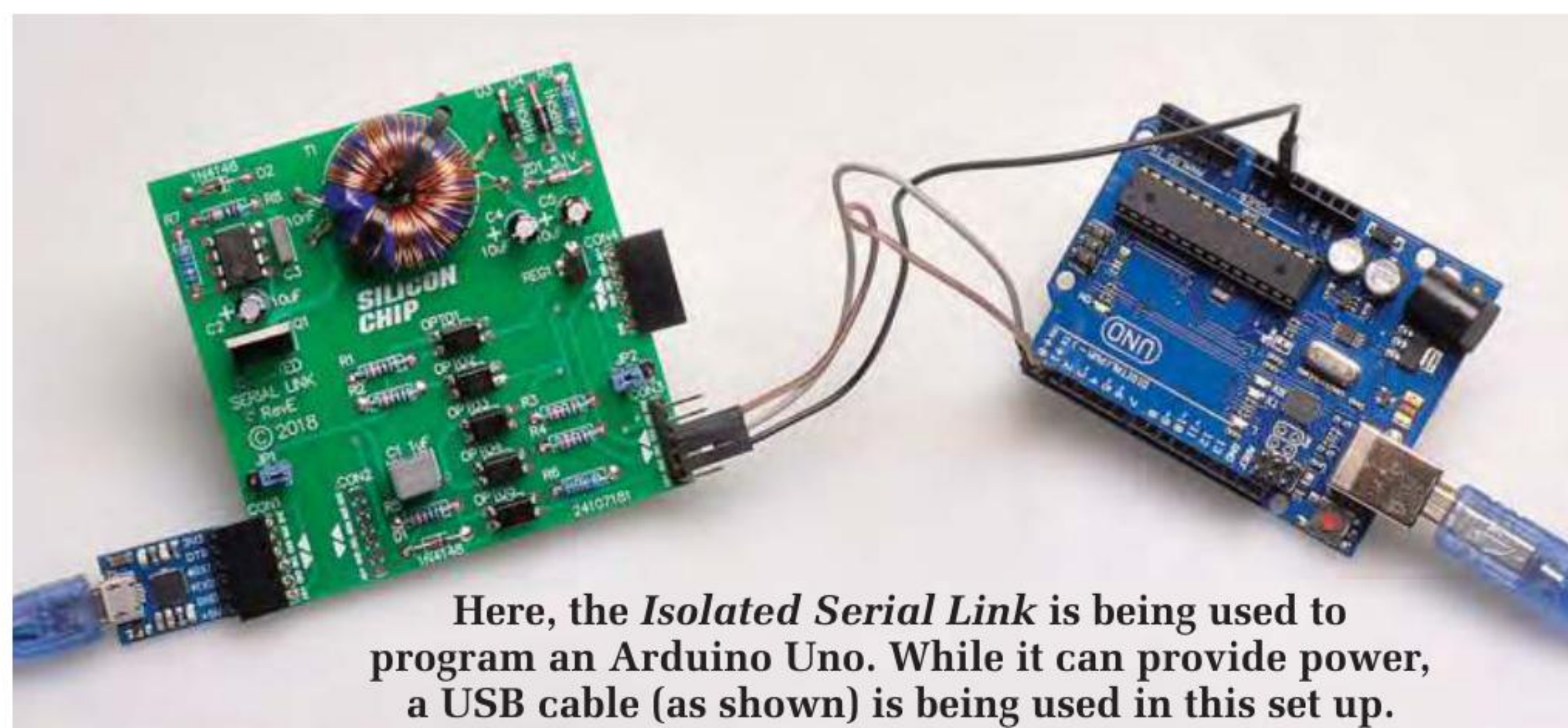
The *Isolated Serial Link* provides two optoisolated data lines suitable for full-duplex serial data (ie, simultaneous sending and receiving). Typically, these connect to the TX and RX pins of a microcontroller or USB/serial converter, although they could be used to pass just about any logic-level signal with a switching frequency up to a bit over 100kHz.

There is also a third isolated data line which can be used to get an Arduino (or similar) board to go into programming mode, so that new firmware data can be sent over the *Isolated Serial Link*. The board includes circuitry to automatically generate a reset pulse on the target board when required.

The board also has an optional small isolated power supply, capable of providing up to about 100mA at 5V. This circuit is based around a 555 timer IC driving a MOSFET, which in turn drives an isolating transformer. This supply can power some (but not all) Arduino boards without the need for a separate power supply. 5V and 3.3V isolated outputs are provided, to suit various situations.

Alternatively, you can omit the 555 IC and transformer (and associated components) and instead mount a pre-built 5V isolated DC-DC converter module capable of delivering up to 200mA (at 5V and/or 3.3V).

While this module can provide more current, it is a specialised part



Here, the *Isolated Serial Link* is being used to program an Arduino Uno. While it can provide power, a USB cable (as shown) is being used in this set up.

(compared to the generic parts used in the transformer-based power supply) and its pins are quite close together, so despite its 1kV isolation rating, it cannot physically provide the same degree of isolation that the transformer does.

Features and specifications

- Provides a fully electrically isolated, bi-directional serial link
- Galvanic isolation up to several hundred volts
- Baud rates up to 115,200
- 3.3V or 5V signalling at either end
- USB/serial interface module can be fitted at either end
- Powered from 5V (eg, a USB port)
- Can be built with isolated 5V and 3.3V supplies for the remote end
- Two isolated power supply options, either 100mA total or 200mA total

Making connections

To connect the external circuitry, the board has two connectors at the left and two at the right. All four have pin-outs that match the ubiquitous CP2102-based USB/serial module (eg, <http://bit.ly/pe-jan20-2102>). One of the two headers on the left side usually interfaces with one of these USB/serial modules to connect to a computer.

This module can be plugged in or permanently soldered to the board, depending on your requirements.

On the other side of the board, one of the communications headers will accept a second CP2102 USB/serial module while the other can be used to plug in where a CP2102 module would. Alternatively, you can just wire up the RX/TX/GND serial connections using jumper leads.

Why do I need one?

The advantage of having an *Isolated Serial Link* is that it allows bidirectional digital communications with no electrical path for current to flow between the two halves of the circuit. This can be handy if the two sides might be at different voltage levels, whether fixed or changing.

For example, let's say that you have designed a circuit which uses a microcontroller and some other circuitry, which is powered from the mains using a 'transformerless' power supply with a current-limiting capacitor feeding a rectifier. This is quite common; many of our designs use this arrangement. But this makes it very difficult to debug your software since you have no way of getting feedback on what's happening in the microcontroller while it is powered from the mains; at least, not safely.

Besides the risk that you could have an accidental short between the incoming mains live and a (supposedly) low-voltage connection, even something as simple as a mains cord or socket with swapped live and neutral wires (which is not uncommon!) could create a lethal situation.

Now though, with the *Isolated Serial Link*, you can safely get serial data from the microcontroller, even if it's floating at mains potential, so you can see what it's doing.

It isn't just mains circuits where it's useful either. For example, you might have a microcontroller with its positive rail connected to the positive terminal of a battery; for example, to simplify monitoring the current drawn from that battery via a high-side shunt.

If the battery bank is earthed, you can't connect to the micro in the usual manner, as you will short out the batteries (and that's a big no-no!). But if you connect it via the *Isolated Serial Link*, that is no longer the case and you can communicate with and re-program that micro as usual.

An *Isolated Serial Link* can even be useful if both devices are nominally at the same potential. If a circuit has more than one ground connection, there is the potential for a ground loop which can cause electrical noise, possibly interfering with the integrity of the serial data or other signals in the circuit.

The *Isolated Serial Link* avoids the introduction of an extra ground connection, thus eliminating the possibility of any ground loops being caused by the serial connection.

Isolating high-speed digital signals

The usual method of optocoupling a digital signal is to apply the incoming signal to the optoisolator's internal LED via a current-limiting resistor, then connect the output transistor either as an emitter-follower or as a common-emitter amplifier, with a pull-up or pull-down resistor respectively.

The common-emitter version of this method is shown in Fig.1. When the input signal goes high, the internal LED switches on and the light it produces causes the output transistor to switch on, connecting the output to VccB and so pulling it high. When the input signal goes low and the LED switches off, resistor RL pulls the output signal line low, to GndB. Since the output transistor is actuated by light, clear plastic between it and the LED provides a high degree of electrical insulation while still allowing signals to travel from one side to the other.

But there is a problem with this configuration: the output arrangement is not symmetrical – the transistor pulls

the output up much faster than the resistor can pull it down. You can use a lower resistor value to speed it up, but that increases current consumption and you can only lower it so far before you overload the output transistor.

A scope grab of this configuration operating is shown in Fig.2. The

input signal is yellow and the output signal is green.

You can see how the pulse is stretched due to the slow switch-off time, despite a relatively low resistor value of 220Ω being used (drawing nearly 25mA when the output is high).

This signal distortion will prevent the receiving end from decoding the serial data above a particular data rate. The fastest baud rate we could achieve reliably with this arrangement was 19,200 baud.

The common-emitter version of this circuit would suffer from the opposite problem, ie, a slow switch-on, resulting in short pulses ('runs'). The outcome is the same: high-speed serial data will not pass through such a link.

A better method

To solve this without resorting to specialised high-frequency optoisolators, we are using pairs of optocouplers in a totem-pole configuration, as shown in Fig.3. One pulls the output high and the other pulls it low. That gives fast, symmetrical drive with a much-reduced supply current.

When the input signal is high, the upper LED is forward-biased and so current flows from VccB, through its output transistor and to the output signal line, quickly pulling it up. And when the input signal is low, the bottom LED is forward-biased and so its associated output transistor quickly pulls the output signal line low, to GndB.

Fig.4 is a scope grab of this type of circuit in operation, and as you can see, the rise and fall times are now essentially symmetrical. While there is a delay of around 5μs, this will not affect serial decoding as the critical logic level thresholds are delayed consistently.

The 115,200 baud limit of this type of circuit is because the delay starts to extend into the next bit time, and at 230,400 baud (the next standard baud rate), the bits are just over 4μs wide, meaning the bits overlap and distort.

The only case where this delay might be a problem at lower baud rates is if the outgoing data is synchronised with the incoming data, either through system design or perhaps a carrier sense bus

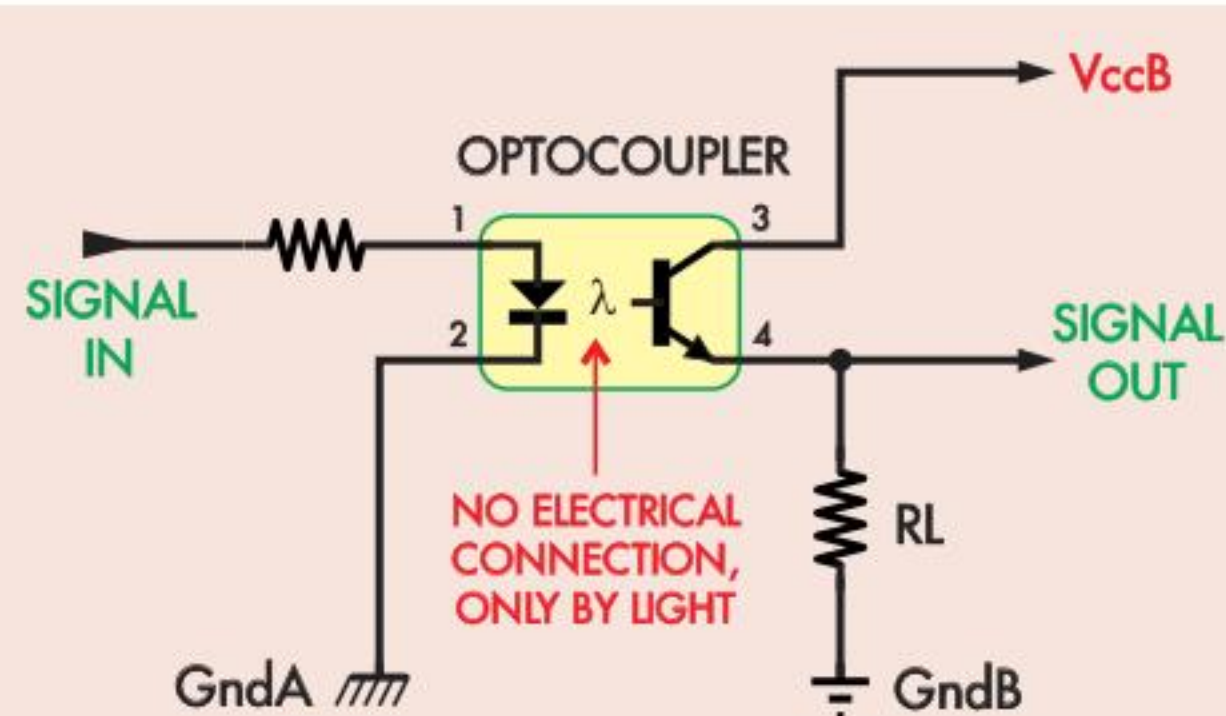


Fig.1: the traditional method of optoisolating a digital signal. When the input signal is high, current flows through the series current-limiting resistor and LED, lighting up the phototransistor, which pulls the output high. But using a resistor to pull the output low when the phototransistor switches off severely limits switching speed, allowing it to handle serial signals up to only 19,200 baud.

arbitration design, where the transmitter is listening in on the receiver to see that it has full control of the bus. But that's a rare situation. For normal serial communications, the delay doesn't matter.

By the way, while it might appear that there is a risk that the supply rails could be shorted out if both optocouplers are switched on simultaneously (eg, with an open-circuit input), phototransistor current is limited to around 20mA by the light intensity generated by the LEDs.

With the input floating, the current through the phototransistors is around 4mA, which is insignificant.

Circuit description

The circuit diagram for the *Isolated Serial Link* is shown in Fig.5. Connections are made to the *Isolated Serial Link* at one end via either CON1 or CON2, and at the other end via either CON3 or CON4.

We'll explain the reason for the pairs of connectors later. At the moment, it's easiest to ignore CON2 and CON4, and just consider the signals and power flow between CON1 and CON3.

Both ends are essentially interchangeable except for the power flow; CON1 receives 5V power to operate the circuit, between pins 1 and 2, while CON3 delivers 5V and 3.3V to any connected circuitry, at pins 1 and 6.

Power flows across the isolation barrier from left to right either via transformer T1 or isolated DC/DC converter module MOD1, depending on which is fitted.

Serial data signals pass in both directions in the manner described earlier, using optocouplers OPTO1-OPTO4.

Data delivered to pin 3 of CON1 appears on pin 3 of CON3 and data delivered to pin 4 of CON3 appears on pin 4 of CON1.

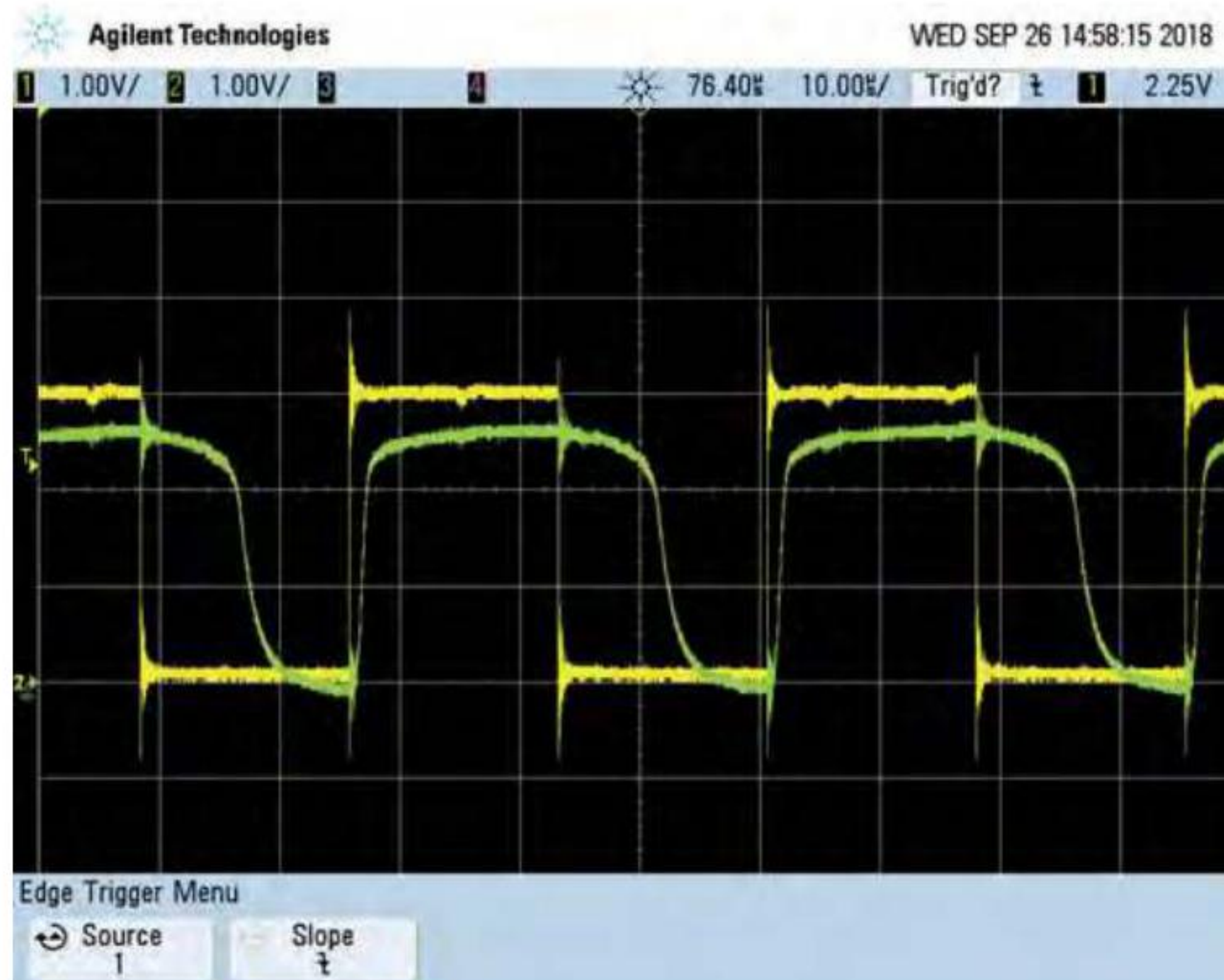


Fig.2: a scope grab showing the operation of the circuit in Fig.1. The yellow trace shows the input signal and the green trace, the output. Note that the output rise time is much shorter than the fall time, which stretches the length of the output pulse. The higher the signal frequency, the more this affects signal integrity.

While these connectors can be wired to just about any circuit which uses TTL serial communications, the pinouts are specifically designed to suit the cheap and readily available CP2102 USB/serial bridges. So you can solder or plug such a device at either end of the circuit to provide a USB interface.

You can choose whether the serial signals at either end have a 3.3V or 5V swing, to suit the type of device that you're connecting. This is selected on the CON1 side using JP1, and with JP2 for the CON3 side.

Note that when you have one side operating at 3.3V and the other at 5V, the optoisolator drive currents are not the same in both directions, but we haven't found this to be a problem – after all, you're usually applying the signal to a digital input pin on an IC, which has a very high input impedance.

Reset signal for micro programming

The fifth optoisolator (OPTO5) is used to pass the DTR flow control signal from CON1 to CON3.

This is often used with Arduino boards, to reset the micro and put it into bootloader mode, so that the chip can be reprogrammed without any additional user intervention.

The DTR signal from a USB-Serial converter is high when the device is idle and no communication is occurring and goes low for the duration of a transmission.

It does not just pulse low when data is being transmitted but is usually held low any time an application has the serial port open.

On typical Arduino boards, an RC network converts the DTR positive-to-negative transition from its onboard USB interface into a brief reset pulse. But this connection is not 'broken out' for use with external serial ports or USB/serial converters. However, the RESET pin connection is available, so

if we can generate this reset pulse from DTR, we can provide the same function.

That's precisely what D1 and its associated 10nF capacitor and 220Ω resistor do. Normally, with DTR high, the 10nF capacitor is discharged and the DTR pin on CON3 (pin 5) is held high by a 10kΩ pull-up resistor.

If the DTR pin on CON1 is externally pulled low, this pulse is coupled through the 10nF capacitor and it powers OPTO5's internal LED. Its associated phototransistor conducts, pulling the DTR pin on CON3 low briefly.

So, if this is connected to the Arduino (or other micro's) RESET line, the micro will be reset.

C1 charges up quite quickly and so after a short time (around 1-2ms), OPTO5 turns off and the RESET line is released.

This is shown in scope grab Fig.6, with the DTR pin of CON1 shown in yellow and the DTR pin of CON3 in green. You can see that both traces drop to 0V around the same time, but the green trace returns to a high level shortly afterwards.

The reset pin on an Atmega328 micro (as used in an Arduino Uno) only needs to be low for 2.5μs to guarantee a reset, so this pulse is more than adequate.

Note that this pulse must be shorter than one second for the programming sequence to complete correctly. D1 discharges the 10nF capacitor when the DTR pin of CON1 goes high again.

If this function is not required and you want to pass the DTR signal through the isolation barrier unaltered, simply replace the 10nF capacitor with a wire link.

Isolated power supply

We've provided two means of getting power 'across the gap'. The simplest approach is to use a self-contained, isolated DC/DC converter module (MOD1), which has a 1kV isolation rating.

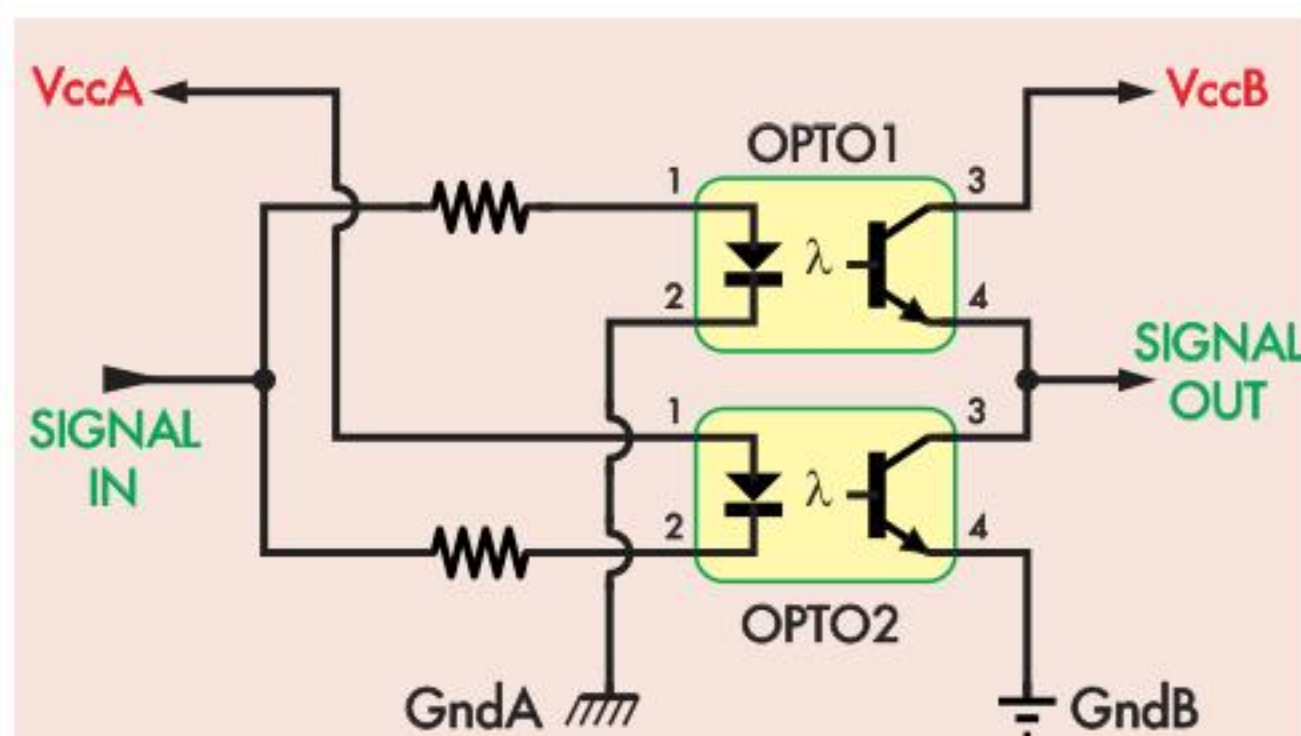


Fig.3: the push-pull digital optoisolator configuration which we're using; a symmetrical arrangement of two optoisolators in a totem-pole configuration. When the input signal is high, the upper optocoupler conducts, pulling the output signal high. A low input signal activates the lower optocoupler, pulling the output low. This will pass a serial stream of up to 115,200 baud.

In this case, the components in the blue shaded box at the top of Fig.5 are not needed. The 4.7 μ F capacitor at left bypasses its input supply while REG1's 1 μ F bypass capacitor provides some output filtering for the module. REG1 at right is a 3.3V low-dropout regulator which provides a 3.3V rail for any circuitry connected to CON3 or CON4.

This is included since many USB/serial converters also provide a 3.3V supply and it's useful for powering certain microcontrollers or other circuitry.

As an alternative to using this module (eg, if you have trouble obtaining it), we have included the circuitry in the box at the top of Fig.5, which comprises a complete isolated power supply. If these components are fitted, you do not need MOD1.

Timer IC1 is configured as an astable oscillator and it drives the gate of MOSFET Q1, which sinks current through the primary of transformer T1 in brief pulses. These induce pulses of current through the secondary winding, which are rectified using schottky diodes D3 and D4 to produce a ~6V DC rail, which is then regulated to 5V by zener diode ZD1.

IC1 uses pretty much the traditional method for a 555-based oscillator except that we've added diode D2 from pin 7 (discharge) to pins 2 and 6 (trigger/threshold) to reduce its duty cycle below 50%.

That's necessary to limit the output voltage at T1's secondary to an appropriate level and also to keep current consumption reasonable (around 100mA). IC1 uses a 10nF timing capacitor, which is charged up via the 820 Ω resistor and diode D2 when IC1's output pin 3 is high.

IC1's off time is determined by the 2.4k Ω resistor and 10nF capacitor, as

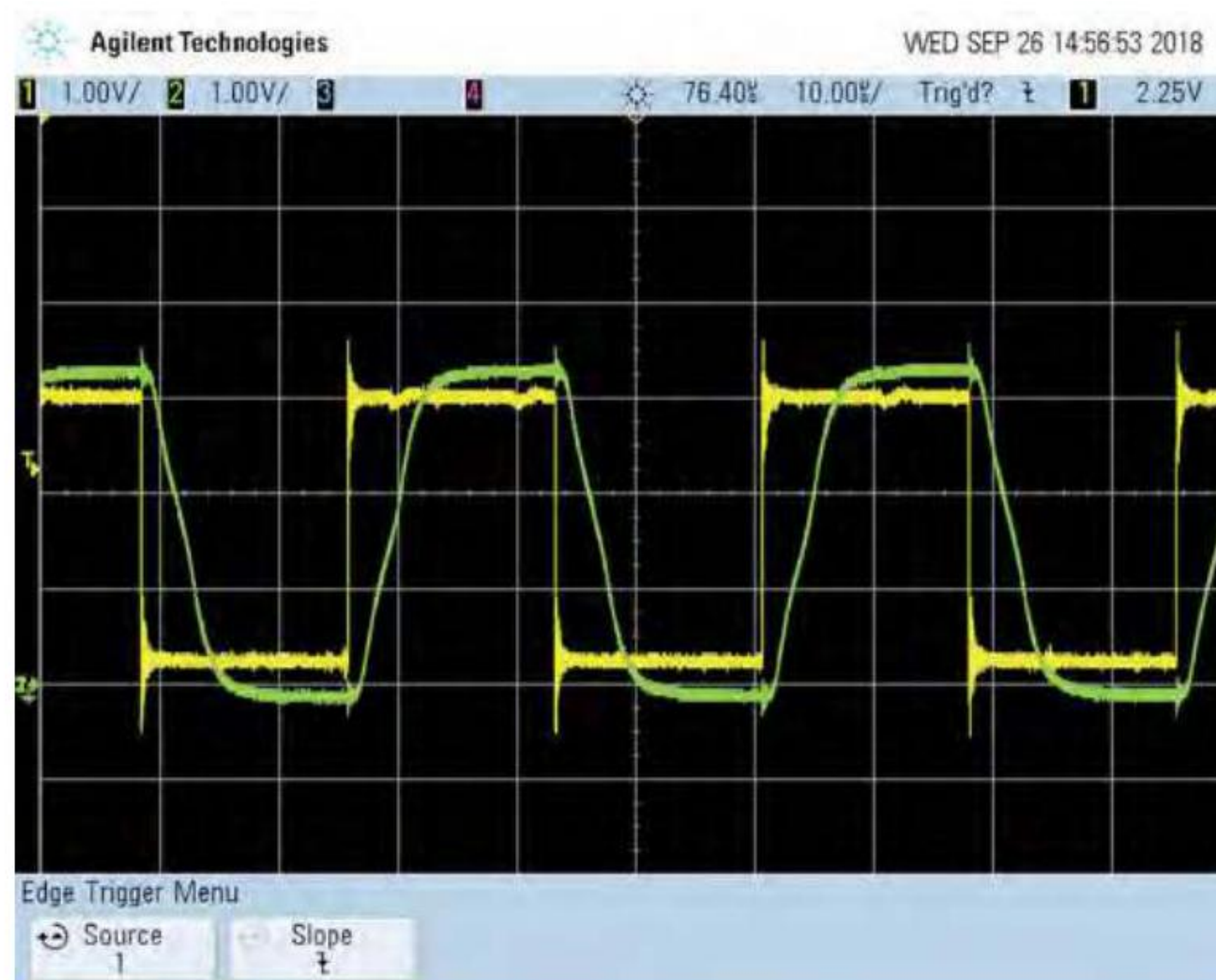


Fig.4: a scope grab showing the same signal as Fig.2 but using the coupling circuit shown in Fig.3. While there is a slight delay between the incoming and outgoing waveforms, the rise and fall times are now similar and short, so the signal can be properly decoded by the receiver at the output end.

the discharge pin (pin 7) goes low when output pin 3 is low, so the 820 Ω resistor and D2 do not affect the capacitor discharge rate.

The result is a square wave with a high period of around 8 μ s and a low period of around 15 μ s, giving a frequency of about 40kHz.

Schottky diodes are used to rectify the output of transformer T1 to minimise power loss, as they have a lower forward voltage and faster switching than typical silicon types. The 5V rail at the cathode of ZD1 is not only fed to the 5V output pins of CON3 and CON4, but also to the input of regulator REG1, which as mentioned earlier, supplies the 3.3V output pins on CON3 and CON4.

Connector options

CON1 and CON2 are wired identically. They are both included to provide you with different options for making connections to the board. You would generally fit one or the other, but not both.

CON1 is near the edge of the board and you can fit a male or female six-pin header which may be vertical or horizontal (eg, using a right-angle type).

We recommend fitting a horizontal female header socket to CON1. You can use a vertical socket but bend its leads by 90° before fitting it. It is then possible to plug a CP2102 module fitted with a right-angle male pin header into this socket (see below). Or you can plug jumper leads into the header.

CON2 is placed further inboard and can be used when mounting a CP2102 USB/serial converter module directly on the board. In this case, you would fit a vertical header and then solder the CP2102 module on top.

Turning now to CON3 and CON4 on the opposite side of the board, these are arranged slightly differently. For a start, they are reversed compared to each other but also, the TX and RX

pins are reversed between them. So if you fit a male pin header to CON3 (ideally a right-angle type), it has the same pinout as a CP2102 module. So by fitting a CP2102 to CON1/CON2, the *Isolated Serial Link* is essentially 'transparent' and you can treat it as a simple CP2102 module, but with the added isolation layer.

You could also fit some other type of header to CON3 and wire it up to another board using jumper leads.

Alternatively, you can fit a female socket for CON4 (right-angle preferred), you can then plug a CP2102 module in, potentially giving you a USB socket at both ends of the module.

That is why the TX and RX pins are reversed; the two sides can then communicate with each other normally.

This is a bit like the old 'null modem' cables (remember them?) that allowed two computers to communicate via their serial ports.

Note though that if you do fit a CP2102 to the right-hand side of the module, it will provide the 3.3V and 5V supplies, so you should leave out all the power supply circuitry on the *Isolated Serial Link* board (including both T1 and MOD1) so that they do not try to 'fight' each other.

Because the DTR/RST signal is not useful in this configuration, it isn't connected to CON4 at all. It's up to you whether you want to leave D1 and its associated capacitor and resistor off the board, since they won't be used.

Winding the transformer

Since we couldn't find a suitably small transformer for T1, we decided to make one ourselves, starting with a pre-wound inductor, which forms the primary. The secondary is then wound on top.

If you are building the unit with the isolated DC/DC converter module, you can skip to the next section.

Start with a 3A or 5A 100 μ H toroidal inductor (we used Jaycar Cat LF1270). Take a roll of electrical tape and cut it into lengths of approximately 250mm, then cut those in half lengthwise, so you have two thin strips.

Wind those strips around the inductor with a slight overlap, forming a complete isolation barrier over the windings, except for two small areas where the leads emerge.

Next, cut a 2m length of 0.4mm diameter enamelled copper wire. It's important to start with the correct length; if it's too short you won't have enough wire, and if it's too long, it will be difficult to wind.

If you start with a different type of inductor then you may need to wind

a different number of turns and will, therefore, need a different length of wire. The number of turns you add should match the number of turns already on the inductor (which will become the primary winding).

Start winding on the opposite side of the core to the existing leads, so that the tails will match up with the pads on the PCB. Leave about 25mm of free wire to connect to the PCB, then wind 50 turns on top of the existing windings, keeping them as tight as possible.

The direction of winding is unimportant, as the output is rectified.

When finished, cut the remaining wire to match the 25mm initial length, then scrape about 5mm of the enamel off the ends of the two leads and tin them.

PCB assembly

Fig.7 shows where to fit the components on the PCB for the version using the transformer to pass power across the isolation barrier. If you are building the version that uses the DC/DC converter module, refer to Fig.8 instead.

Fig.9 shows how to assemble the PCB if you have 3.3V or 5V DC power available at both ends of the *Isolated Serial Link*.

All three versions are built using the same PCB, which is coded 24107181, measures 74 $\times$ 74mm and is available from the *PE PCB Service*. The following instructions describe fitting all the parts; ignore the instructions to fit any components which your version does not require.

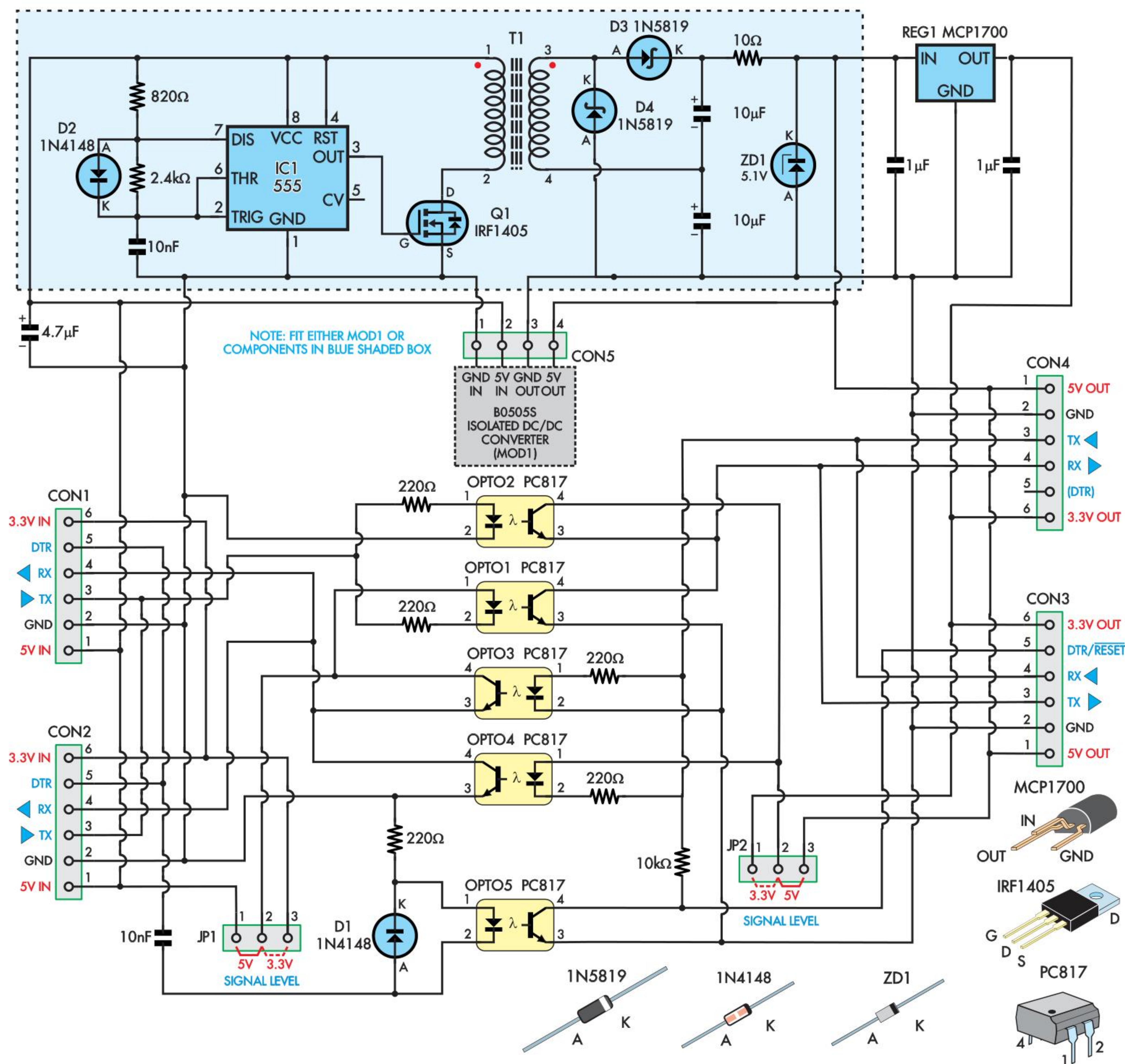


Fig.5: this circuit of the *Isolated Serial Link* has the optional isolated power supply at the top (blue box). The alternative isolated DC/DC module which can be used instead is near the centre (grey box). The isolated bidirectional serial data link is provided by OPTO1-OPTO4. OPTO5 couples the DTR signal from left to right.

Start by soldering the resistors in place. It's a good idea to check each value using a multimeter before fitting them, as the colour bands can be difficult to read. Be sure to trim all the leads neatly after soldering, as stray leads left over could potentially compromise the isolation barrier.

Mount the diodes next. D1 and D2 are small 1N4148 types, while D3 and D4 are larger schottky diodes. They are all polarised, so check that each cathode band is facing as shown on the relevant overlay diagram before soldering it in place. Note that D3 and D4 face in opposite directions.

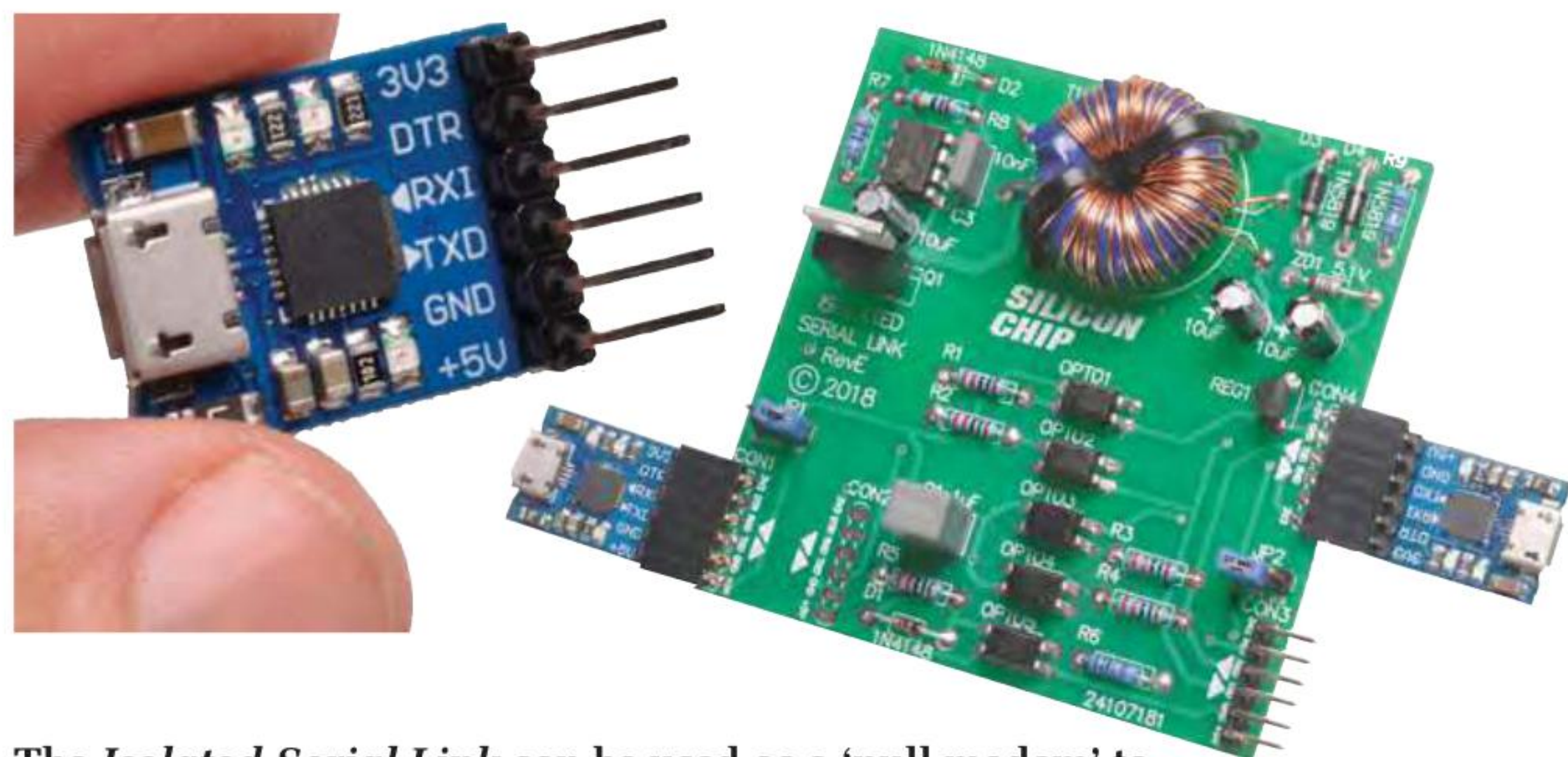
There is also one zener diode, ZD1, and now is a good time to fit it, with the orientation as shown. The five optoisolators can be mounted next. They are not all oriented the same way. OPTO1, OPTO2 and OPTO5 have their pin 1 facing the top of the board while OPTO3 and OPTO4 have the opposite orientation.

Line up the dots and notches on the optoisolators with the PCB and ensure they sit flush before soldering all the pins.

The MKT and/or ceramic capacitors are next on the list. These are not polarised. Install them where shown, then mount small regulator REG1 with the orientation shown. You will need to bend its leads to suit the PCB pad pattern (eg, using small pliers).

Now you can fit the electrolytic capacitors, which are polarised. The longer lead is positive, so feed it into the pad marked with a '+' in each case. The stripe on the can is on the side with the negative lead. IC1 can be soldered directly to the board (preferred) or mounted using a socket.

Regardless, the notch in IC1 and the socket should face towards the bottom of the PCB. You may need to straighten the IC legs slightly so that they fit through the holes in the PCB or into the socket.



The Isolated Serial Link can be used as a 'null modem' to allow communication between two computers. Note that two USB/serial converter modules are needed for this application. Since each side is supplied with power, the power transfer circuitry is unnecessary.

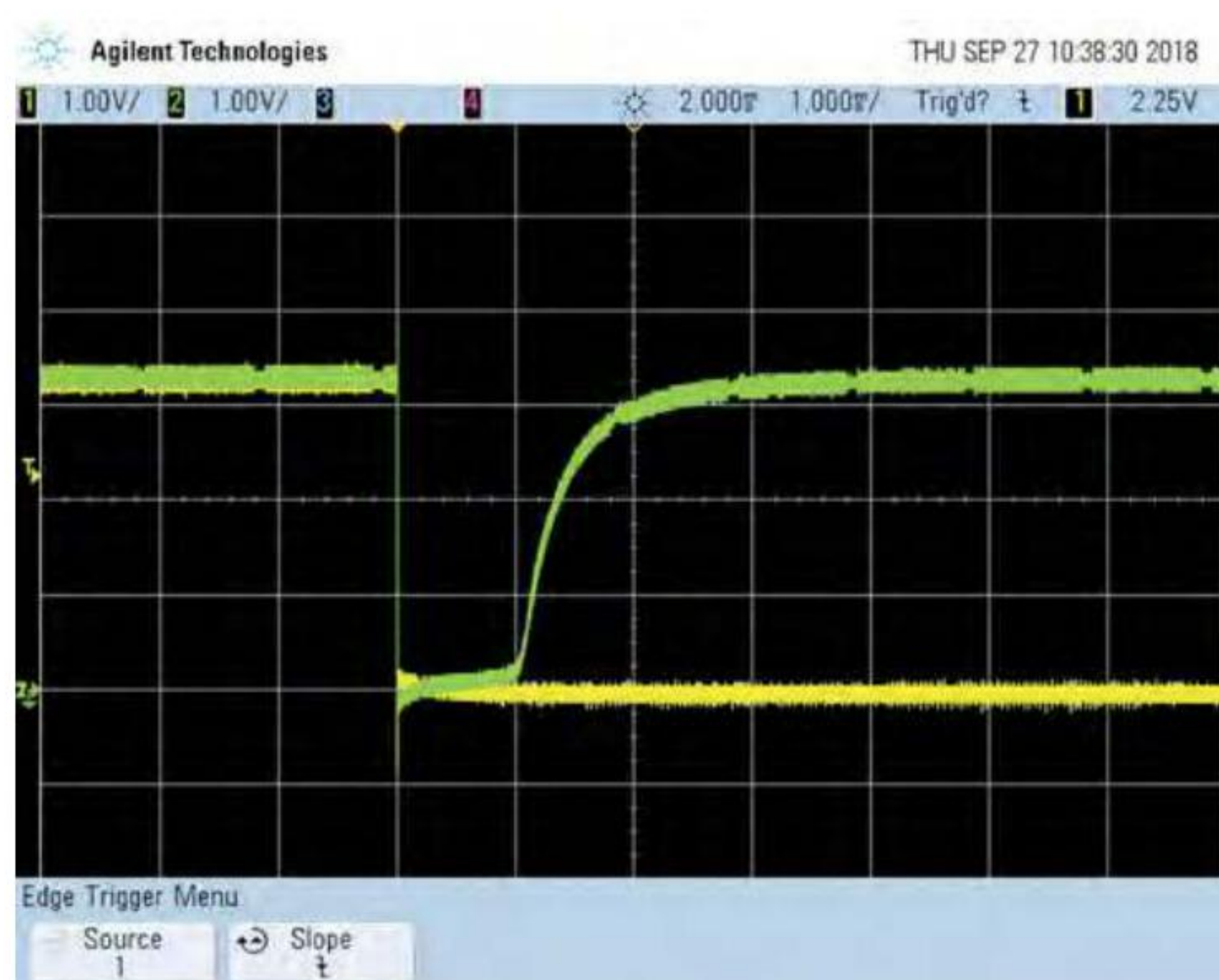


Fig.6: the yellow trace is the signal applied to the DTR pin of CON1 while the green trace shows the signal at the DTR/RESET pin of CON3. The falling edge of the DTR input generates a 1-2ms low pulse at the reset output, which can be used to reset the microcontroller in an Arduino-compatible board, activating its bootloader and allowing the chip to be reprogrammed.

Next, fit the sockets for CON1-CON4. The exact arrangement used will vary depending on how you are planning to use the unit. If you are not sure, fit all the sockets as shown in our photos and on the overlay diagrams and then you have various options later. Figs.7-10 show some examples of various ways to use the board. At the same time, solder the two 3-pin headers for JP1 and JP2 to the board.

Solder the primary windings (made with thicker wire) to the pads on the left-hand side of transformer T1 with the thinner secondary connections on the right. Secure the transformer to the board using two cable ties, through the holes in the PCB.

If fitting DC/DC converter module MOD1, line up its outline with the footprint marked on the PCB, noting that the leads are closer to one edge than the other. The component markings should face towards the middle of the PCB. Solder it in place, keeping it flat and level.

Now mount Q1 with its metal tab facing towards the top of the PCB, as shown. If you like, it can be bent forward to sit parallel to the PCB. In this case, the tab will face up. No heatsink is required.

Using it

Before plugging it in, install the jumper shunts for JP1 and JP2 to match the voltage of the serial signals that will be applied to each side of the board. We found the 5V selection to work best for CP2102 USB/serial modules.

If in doubt, test the voltage of the TX line of the equipment you are planning to connect while it is powered but not transmitting. Serial data lines usually sit at a high level when idle, so this will give you an accurate reading of the voltage level.

Typically, you would connect a computer or other device which can supply power to run the circuit to the left-hand side of the unit (via CON1 or CON2).

If you have installed either T1 or MOD1, the unit can supply a modest amount of power to devices connected to either CON3 or CON4, up to around 100mA at 5V. This is enough to power something like a bare Arduino board but it will be overloaded if you try to power a board with a lot of extra accessories such as an LCD screen or motor.

In this case, you can power the circuit at the 'remote' end using a battery pack, keeping in mind that if you wish

IMPORTANT! – Notes for mains use

When using the Isolated Serial Link for isolating circuitry floating at mains potential, the following precautions must be observed:

- 1) It must be mounted in an Earthed metal or double-insulated case before connecting it to the mains-powered equipment (ideally, within the same enclosure). Only the isolated connections should be brought outside the case. If mounting in a separate case, the wiring to the mains-powered equipment must be mains-rated and properly insulated at both ends.
- 2) Either omit the isolated power supply circuitry or build the version using MOD1, not transformer T1.
- 3) If using MOD1, lengthen the slot underneath it until it nearly touches OPTO1 (the slot is already lengthened on RevH boards).

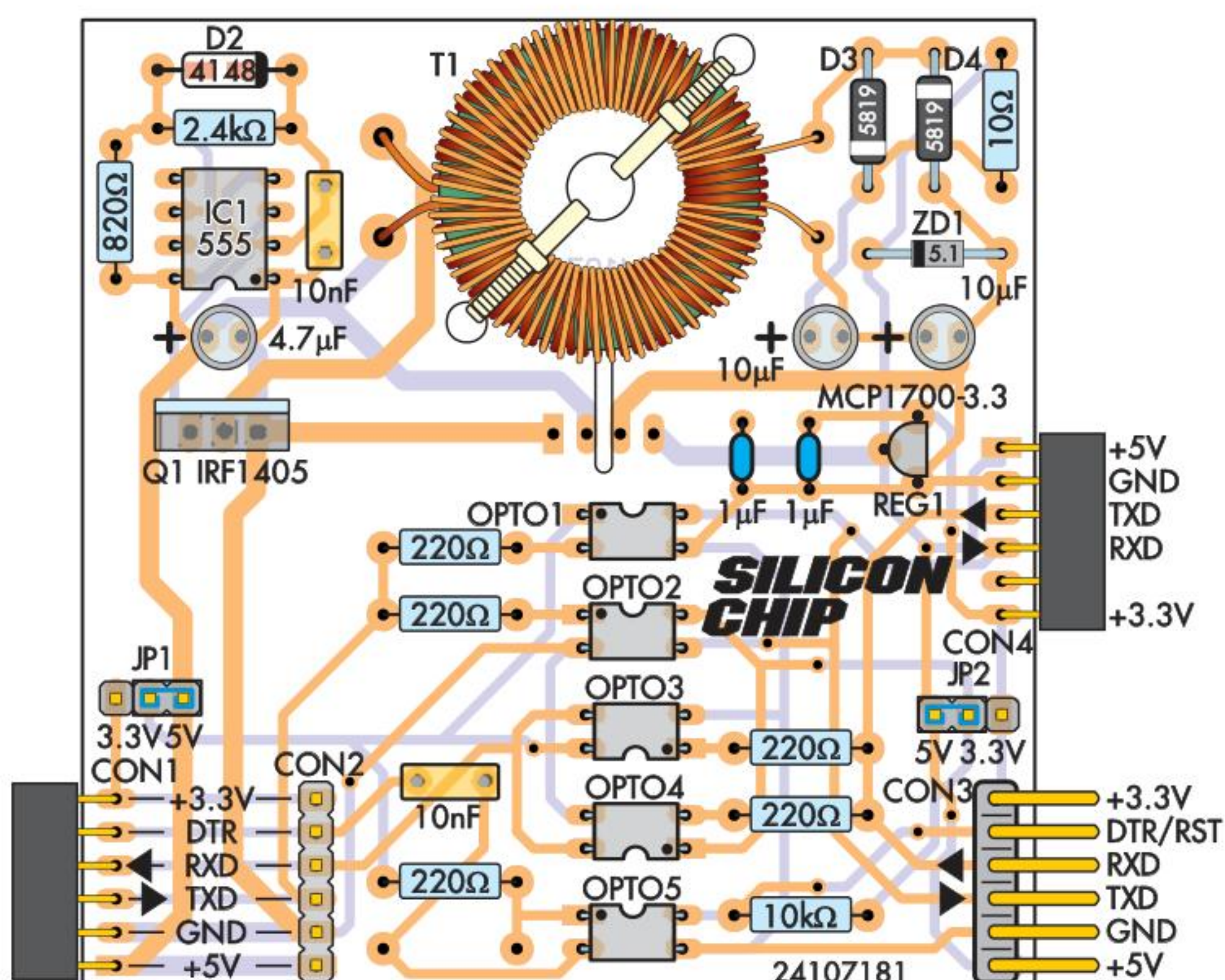


Fig.7: where to fit the components for the version of the *Isolated Serial Link* which uses a transformer to provide isolated power to circuits connected via CON3 or CON4, drawing power from CON1 or CON2. This shows all four connectors fitted but you don't have to fit them all – and you can also use different types, to suit your application.

to maintain isolation, no part of the two sides should be connected. In this case, you only need to make connections to the following pins on CON3/CON4: RX, TX, GND and RST (if needed).

It's always a bit tricky connecting the TX and RX lines between two boards because there are some cases where you connect the pin labelled TX to TX and other times when you connect TX to RX, depending on the labelling scheme used.

So to help remove some of the confusion, we've printed small arrows on the PCB (visible in Figs.7-9) which show the direction of data travel on each pin.

Treating the unit as an isolated CP2102 board

If you have a setup where you would normally use a CP2102 module to communicate with a device but you need isolation, you either plug a CP2102 module into CON1 (female header) or solder it to CON2. CON3 then provides

a more-or-less identical function to the original CP2102 pins except for the added isolation layer.

So if you have a socket which will accept a CP2102 module header, CON3 will have a matching pin-out and can be used as a direct replacement.

Connecting to an Arduino

This is especially helpful if your Arduino is connected to circuitry operating at much more than 5V (especially a battery which can supply a lot of current), or even mains. The isolation barrier will prevent any accidental shorts or component failures on the Arduino or any connected modules from damaging your computer.

In this case, we suggest you use the board with a CP2102 USB/serial module attached to either CON1 or CON2. Run jumper wires from either CON3 or CON4 to the Arduino board, connected as follows: GND to GND, RX to TX and TX to RX.

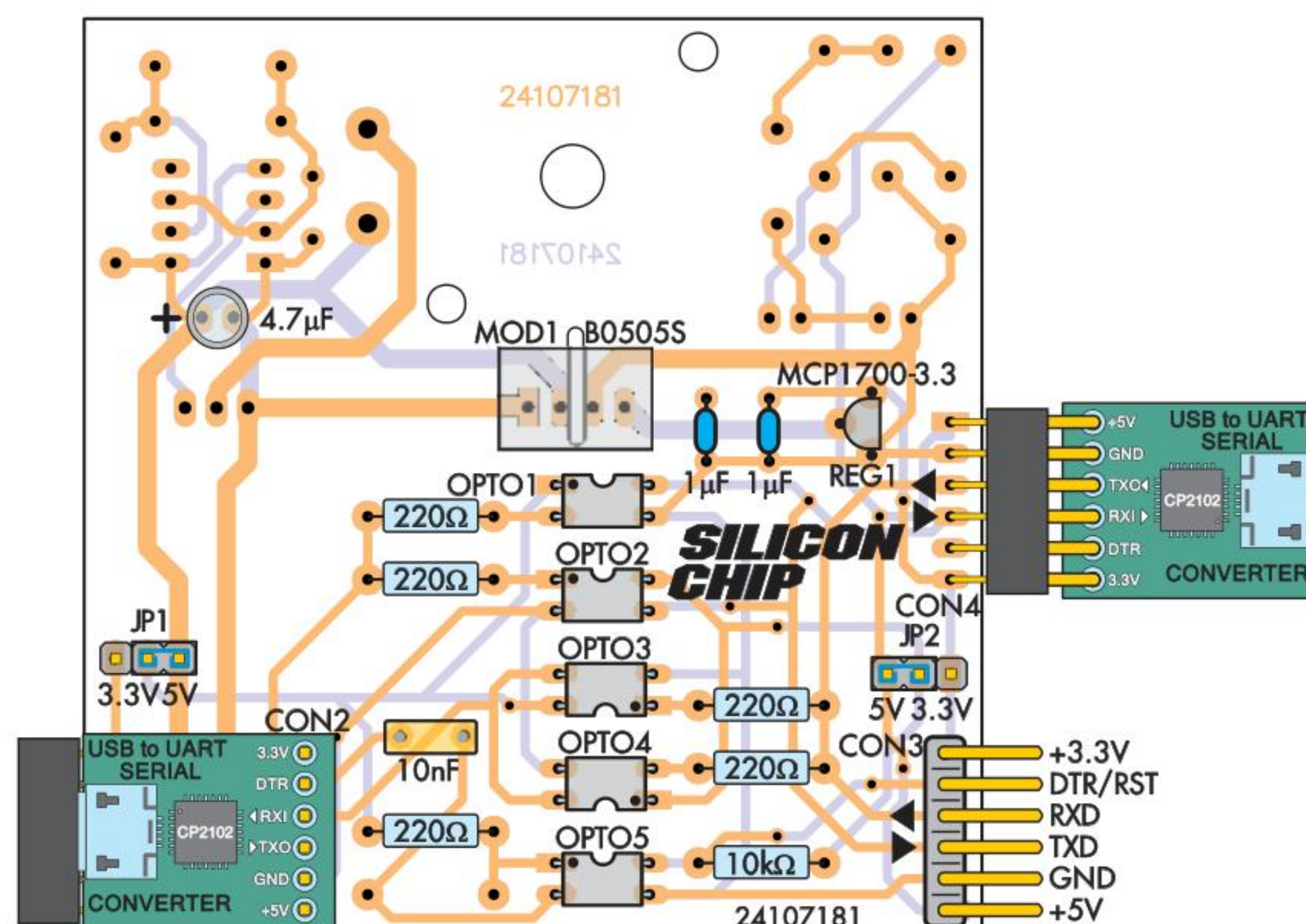


Fig.8: if you build the version with the isolated DC/DC converter module (MOD1) instead of transformer T1 and associated components then only fit the parts shown here. This time we're showing a CP2102-based USB/serial module mounted on the board via CON2 and another plugged into CON4, but that's just an example of how you can use it.

The reason why TX is not connected to TX and RX to RX is that the signal that is being transmitted by one side is being received by the other. This arrangement is shown in Fig.10.

To be able to reprogram the Arduino while it is connected over the *Isolated Serial Link*, you will also need to connect the pin labelled RST on the *Isolated Serial Link* to the RST pin on the Arduino.

Note that this will only work with Arduino boards that communicate via a USB-Serial IC which is separate to the main processor IC.

We have tested this on the Uno and Mega-compatible boards but it will not work with boards such as the Leonardo because they do not expose their serial programming lines directly.

Boards such as the Nano should allow programming, as they use a similar design to the Uno and Mega, although we have not tested this. Other Arduino variants may or may not work, depending on how they are configured.

Note: the power supply built around T1 may be able to supply enough power to the Arduino during programming but it's possible that it can't, as Arduino boards can be quite power hungry, even when doing nothing.

Connecting two computers

To provide an optoisolated link between two computers (or a computer and Raspberry Pi), you will need to connect two CP2102 modules to the *Isolated Serial Link*. Connect one to either CON1 or CON2 and the other to CON4.

Since both computers can supply power, none of the power transfer circuitry is needed. Note that the DTR/



In the absence of a commercial model, we wound our own transformer using a 5A 100μH toroidal inductor. After insulating the winding with tape, we wound on a secondary which matched the number of turns on the 'primary'.



The completed transformer is held in place on the PCB with a pair of small cable ties through the holes provided. Cut the excess from the ties on the underside of the board. (Don't use wires to hold it in place because they could form shorted turns and seriously degrade performance!)

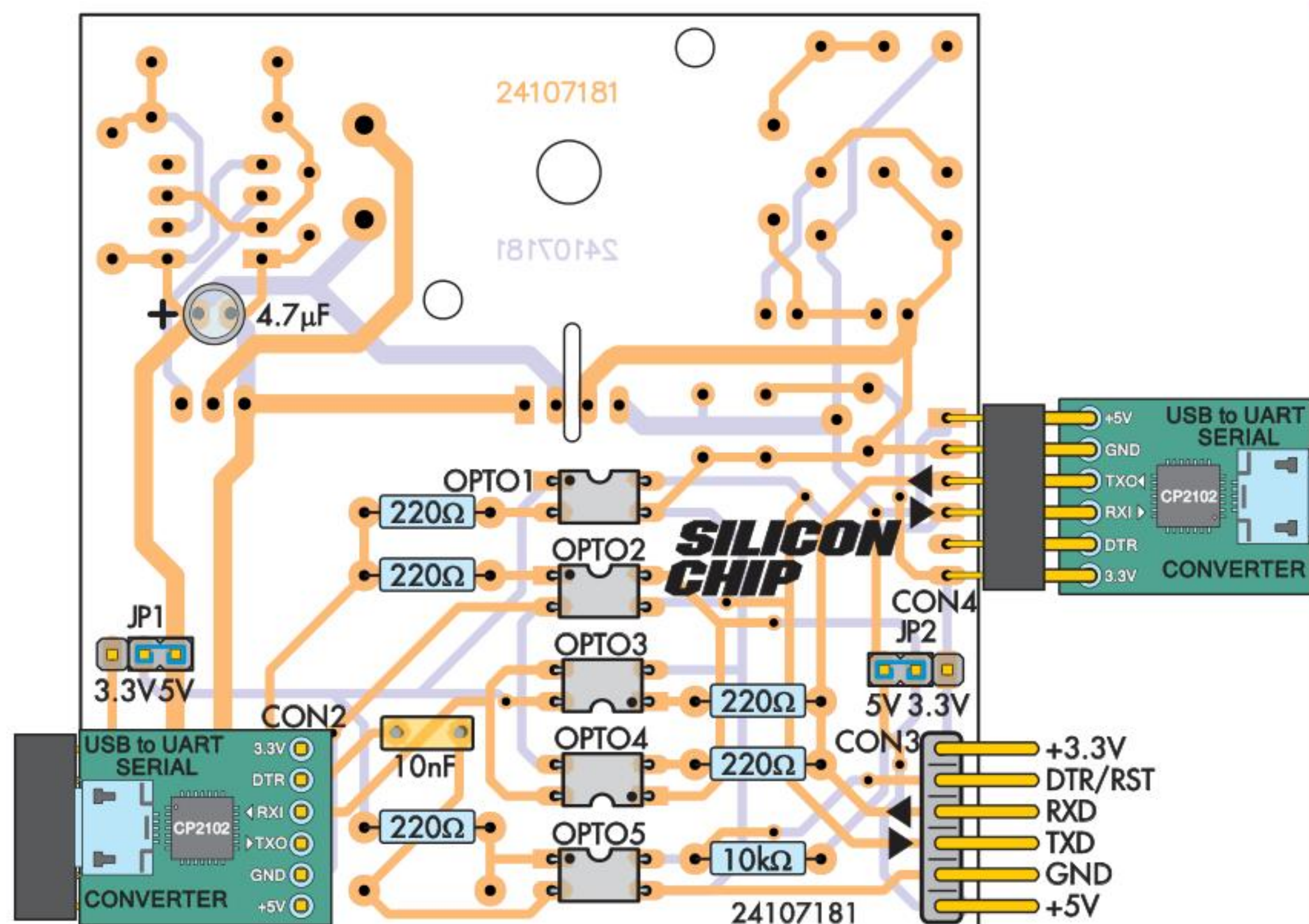


Fig.9: this shows which components you need to install if you're supplying 5V power to both sides of the board, and do not need an isolated supply to transfer power from CON1/CON2 to CON3/CON4. For example, you would use this configuration if you're connecting a USB/serial converter module at both ends, as shown here.

RST signal will not be used either, so OPTO5 and its associated components could be omitted.

Using other USB/serial converters

While the board was designed to suit CP2102-based modules, other types can be used. Note though that this unit has been designed to work with TTL level signals, and will not work with RS-232-voltage-level signals.

Just make sure to set the correct voltages on each side and also connect the correct power and signal connections. Using jumper wires with socket ends onto the pin headers is an easy way to do this.

You can even use a minimal amount of cyanoacrylate glue (superglue) to join the socket ends of the jumper wires together, to create a removable harness.

Reproduced by arrangement with
SILICON CHIP magazine 2019.
www.siliconchip.com.au

Parts List – Isolated Serial Link

- 1 double-sided PCB coded 24107181, 74mm × 74mm, available from the PE PCB Service
- 2 6-pin female headers (CON1, CON4) [Altronics P5374]
- 2 6-pin male headers (CON2, CON3) [Altronics P5430, Jaycar HM3212]
- 2 3-way pin headers with jumper shunts (JP1, JP2) [Altronics P5430 and P5450 or Jaycar HM3212 and HM3240]

Capacitors

- 1 4.7µF 16V electrolytic capacitor
- 2 1µF MKT or multi-layer ceramic
- 1 10nF MKT

Semiconductors

- 5 PC817 opto-isolators (OPTO1-OPTO5) [element14]
- 1 MCP1700-3.3V LDO 3.3V regulator, TO-92 (REG1)
- 1 1N4148 signal diode (D1)

Resistors (all 1% 1/4W metal film)

- 1 10kΩ
- 5 220Ω resistor

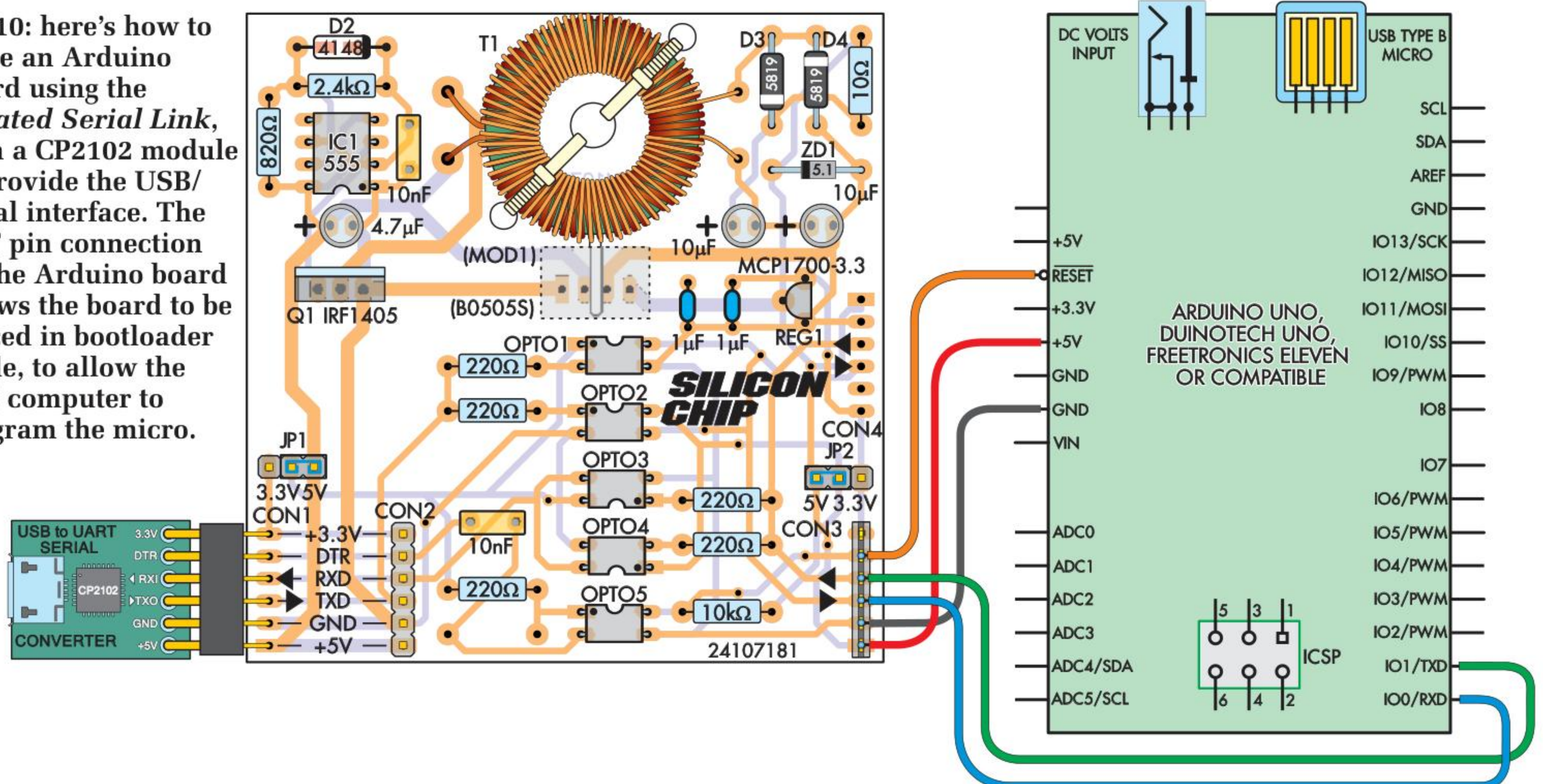
Extra parts for version using MOD1 (optional)

- 1 B0505S-1W 5V-5V DC-DC isolated converter *or* LME0505SC [element14] *or* RFM-0505S [Mouser]

Extra parts for version using T1

- 1 100µH 5A toroidal powdered iron inductor (T1) [Jaycar LF1270]
- 1 2m length of 0.4mm-diameter enamelled copper wire (T1)
- 2 small cable ties
- 1 NE555 or equivalent timer IC, DIP-8 (IC1)
- 1 IRF1405 N-Channel MOSFET, TO-220 (Q1) [Jaycar ZT2468, Altronics Z1545]
- 1 5.1V 1N4733 Zener Diode (ZD1) [Jaycar ZR1403, Altronics Z0614]
- 1 1N4148 signal diode (D2)
- 2 1N5819 1A schottky diodes (D3, D4)
- 2 10µF 16V electrolytic capacitors
- 1 10nF MKT capacitor
- 1 2.4kΩ 1% 1/4W resistor
- 1 820Ω 1% 1/4W resistor
- 1 10Ω 5% 1/2W resistor
- 1 500mm length of electrical tape

Fig.10: here's how to drive an Arduino board using the *Isolated Serial Link*, with a CP2102 module to provide the USB/serial interface. The RST pin connection on the Arduino board allows the board to be placed in bootloader mode, to allow the host computer to program the micro.





by **Nicholas Vinen**

Four-channel High-current DC Fan and Pump Controller Part 2

In the December 2019 issue we revealed our new high-current fan and pump controller, able to switch up to 40A total with a 12V nominal supply, controlling up to four loads using the readings from between one and four temperature sensors. Plus, it's programmed over USB to make the many different settings easy to control. In this second part, we cover PCB assembly, wiring it all up and adjusting those settings to suit your installation.

One of the main goals we set with this new *DC Fan and Pump Controller* was to provide many different options to suit different situations, without making it a nightmare to configure. We certainly couldn't use jumpers and trimpots because there would be just too many, and it would be too hard to make any changes once the unit was mounted in a vehicle.

So instead, we have made the unit configurable and controllable over a USB text interface. Unfortunately, the low-power micro we've chosen doesn't have a great deal of memory, but we've come up with a way to provide a friendly user interface that allows you to see the exact settings and make changes via a laptop or desktop PC.

Basically, you view and change your settings via a web page which then produces a 'magic string' of text which, when pasted into the *Controller's* terminal, changes its behaviour to match up with what you have entered on the web page.

So if you aren't happy with the way your fans and/or pumps are being operated, it's a simple matter to reach the accessible USB plug or socket you've fitted, connect it to your PC and upload a new configuration. You can even test it without having to take the vehicle out on the road, simulating battery voltage and changes, and temperature sensor changes to see what happens.

We've used mostly SMD parts, so the PCB is reasonably compact, thus fitting the *Controller* inside the vehicle is made easier. Despite this, it can be tricky to find somewhere to position it, but we did find a good location in the packed engine bay of our test vehicle, and the wiring is pretty easy, once you've purchased appropriate connectors and got the hang of soldering them. And anyway, it's heaps more capable and configurable than many other designs, so the small penalty in size is well worth it.

PCB construction

The *Fan and Pump Controller* is built on a PCB coded 05108181 measuring 68 × 34.5mm, available from the *PE PCB Service*. All components are mounted on the top side. Use the PCB overlay diagram, Fig.4, as a guide during assembly.

If you are fitting onboard USB socket CON1, start with that. Spread a thin smear of flux paste on its four mounting pads and five signal pins, then drop the socket on the board and move it around until the two plastic posts on the underside drop into the alignment holes. You should find its five pins are then positioned over the pads.

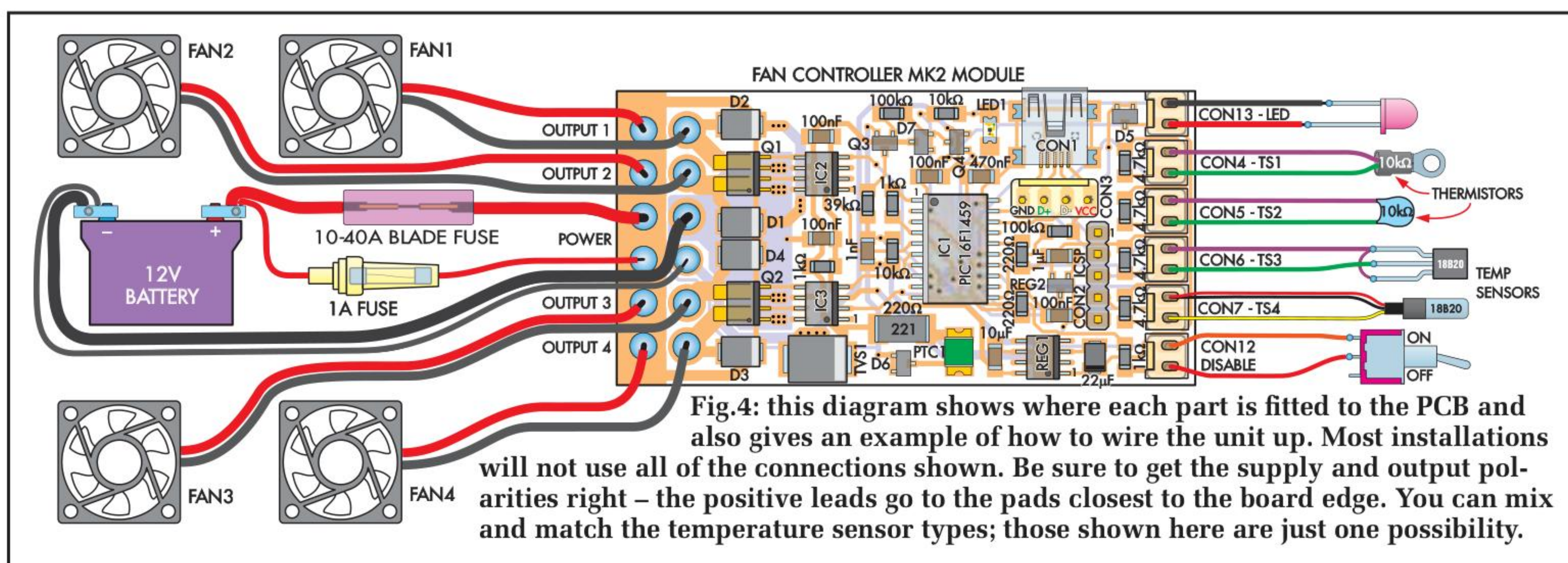
Nudge it a little if necessary, to get the alignment perfect. Then apply solder to one of the four large pads which attach its 'feet'. You will need to apply a fair bit of heat and some extra solder to get a good, solid joint. Re-check the signal pin positions, and if necessary, reheat that solder joint and carefully nudge the part without lifting it up. It may be hot, so use caution.

Once you're happy with the position of the signal pins, solder the other three mounting feet, then apply a small amount of solder to those pins. If you load some solder onto the tip of your iron and touch it to the end of the pin (which is partially hidden under the body), the flux paste you applied earlier should help to 'suck' the solder off the iron and onto the pin and pad.

Repeat this for the other four signal pins and carefully examine them under a magnifier with good light, to ensure a good joint has formed and there are no bridges between pins. If there are bridges, apply a little extra flux paste and then use solder wick and heat from the iron to remove them.

Next, move onto microcontroller IC1. It is in a wide SOIC package with relatively large pin spacings, so it is not difficult to solder. First, find its pin 1 dot and make sure that it is oriented as shown in Fig.4. Also, check that it is sitting flat on the board, then tack solder one of its corner pins. It's easier to solder if you spread a small amount of flux paste on all its pads first.

Make sure all the pins are correctly aligned on their pads. If not, heat that initial solder joint and gently nudge it into position. Repeat until you are happy that they are all lined up, then solder the remaining pins and finally, add a little extra solder to the first pin to refresh the joint. Inspect the joints, and as before, if you find any bridges, clean them up with flux paste and solder wick.



Now you can proceed to solder IC2, IC3 and REG1 similarly, as they are all in smaller SOIC packages. Note though, that their pin 1 dot is oriented differently to IC1. Check the orientation carefully against what is shown in Fig.4 before soldering each chip.

MOSFETs Q1 and Q2 should be fitted next. These are in a similar package to IC2, IC3 and REG1 except that the pairs of pins on one side are joined together. So we have provided larger pads to solder those pairs of pins to the board. Again, check that the pin 1 dot is oriented correctly – the same as IC2 and IC3 – before soldering them in place.

These are seven small three-pin, SOT-23-package parts on the board: Q3, Q4, D5-D7 and REG2. They look similar so don't get them mixed up. Their pins are widely spaced, so they are pretty easy to solder. Use the same technique as with the ICs; it's generally easier to tack the pin that's all by itself on one side first, then solder the other two pins and refresh the first solder joint last.

Now fit the smaller (3216/1206-size) resistors and capacitors. The required values and positions are shown in Fig.4. They are not polarised, so orientation is not important. The resistors will be printed with a 3-digit or 4-digit code on the top to indicate their value, while the ceramic capacitors will be unmarked, so be careful not to mix them up.

It's the same basic method – tack one end, check the positioning and then solder the opposite side and go back and refresh the first joint.

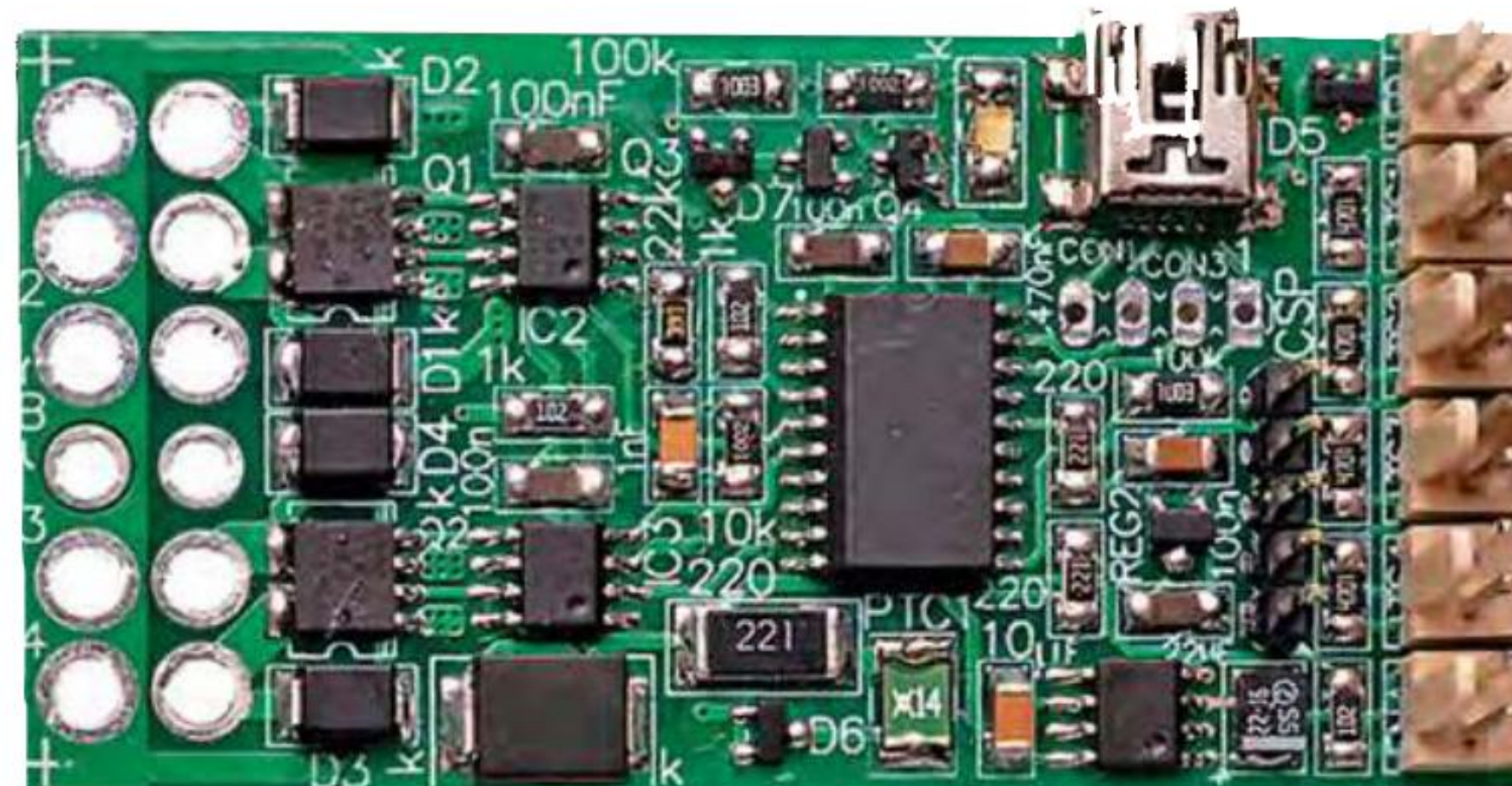
Besides making sure the parts are flat on the board and that the solder joints are made properly, the main trick is to be patient and wait several seconds between soldering one side of the part and the other. This gives the joint time to solidify. Otherwise, the part will tend to move out of position when you touch it with the iron.

You can now fit PTC1 and the large 220Ω resistor next to it, using the same basic technique. Keep in mind that these larger parts will require a bit more heat and solder to form good joints. Neither of these components are polarised.

The diodes are also two-terminal devices and can be soldered in the same manner as the passives but are larger again, so they will also need a bit more heat. Fit diodes D1-D4 now, ensuring that their cathode stripe faces towards the right side (ie, into the middle of the board). You can also fit TVS1 now; it's larger again, but otherwise like the other diodes.

The last remaining SMD component is the 22μF tantalum capacitor next to REG1. It is also polarised and must be soldered with its positive end (generally marked with a stripe) towards the bottom edge of the board.

You can now move on to fit the headers that you require for your application. You will need at least one of the four temperature sensor headers (CON4-CON7); we recommend that you fit all four, even if you aren't planning to use them, in case you want to add more sensors later.



The completed *Fan and Pump Controller* is shown here slightly oversize for clarity (actual PCB size is 68mm wide – as seen above). Yes, it is all SMD components, so a good eye, a steady hand and a fine-tipped iron are all required.

You can also fit CON12 and/or CON13 now, for the enable/disable control and indicator LED. Again, you may want to fit them even if you aren't planning to use them, in case you change your mind later.

Planning the wiring

As mentioned in the first article (December 2019), rather than use connectors for the high-current wiring, we have simply provided large pads on the board, to which fairly thick wires can be soldered directly. While it is possible to use fixed cables, we suggest that you use in-line connectors on most or all of the wires.

This has a few advantages: it makes testing easier, it makes it easier to replace a sensor or fan later if you have to, it makes it easier to remove the unit in case you need to repair or reprogram the unit, and so on.

There are various suitable types of inline automotive connectors, many of which are waterproof. While waterproof connectors are not critical for the 12V supply wiring or connections to fans/pumps, we recommend that you use them for the sensors, enable/disable line and external LED wiring (if used) as water may conduct enough current to affect the function of those devices. See the panel below for more details on suitable connectors that are available.

Having decided where you will have connectors and what type to use, you will then need to find a suitable location for the case that will house your PCB. We strongly suggest that you use an IP65 (or better) rated waterproof box. You could use an ordinary plastic box and waterproof it with silicone but it will be hard to get it apart later if you need to.

We used a sealed ABS plastic box from Jaycar – see the additional parts list (overleaf) for details. Figure out where your box will fit in the vehicle and also how you will attach it. We used a screw through one of the box's two integral mounting holes, through a support member in the vehicle

(which already had a hole in it) and into a piece of foam, capped off by a washer and a nut.

We also placed a thin piece of foam (with a hole in it) between the box and the cross member. This provides some vibration reduction compared to rigidly mounting it to the vehicle.

Now that you have a location for the box, you can measure the lengths of all the required cables.

The easiest way to measure how long a cable needs to be is to thread a spare piece of wire through the vehicle between the two points to be connected, loosely, then pinch the end in one hand, pull it out and measure its length. Remember that some parts of the car may flex or move, so don't make it too tight.

You will also need to calculate the minimum current rating for each. This will typically be 10-20A for fan cables and 10-40A for the battery cables. Just about any wire can be used for the sensor wiring, enable/disable switch, LED and battery voltage-sense wiring, since all of these carry mere milliamps.

When cutting the cables to length, remember to account for the length lost stripping both the inner and (where present) outer layers of insulation, plus a bit extra in case you damage the wire while stripping it and have to cut it off.

Having cut and stripped the insulation off the ends of all the various cables required, crimp and/or solder the connectors on. Leave the connectors that will plug into the PCB off for the moment.

Parts list – Fan/Pump Controller for sample installation with one fan and three temperature sensors (change to suit yours)

- 1 DC Fan/Pump Controller PCB, assembled (see last month)
- 1 IP65-rated sealed high-temperature ABS box, 15x65x40mm [Jaycar Cat HB6122]
- 1 USB mini-B to type-A cable
- 2 30A waterproof blade fuse holders with LED [Jaycar Cat SZ2042]
- 1 1A blade fuse [Jaycar Cat SF2126]
- 1 20A blade fuse [Jaycar Cat SF2138]
- 2 6mm non-insulated eye terminals [Jaycar Cat PT4934]
- 1 4-way Deutsch waterproof plug/socket set [Jaycar Cat PP2149]
- 1 2-way Narva-style waterproof plug/socket set [Jaycar Cat PP2110]
- 1 4-way Narva-style waterproof plug/socket set [Jaycar Cat PP2114]
- 1 2-way 250-series automotive socket (to suit radiator fan) [Jaycar Cat PP2062]
- 1 1m length 2-core 7.5A automotive cable [Jaycar Cat WH3057]
- 1 1m length 2-core 15A automotive cable [Jaycar Cat WH3079]
- 1 1m length 2-core 25A tinned automotive cable [Jaycar Cat WH3087]
- 1 1m length 25A black tinned automotive cable [Jaycar Cat WH3082]
- 1 1.2m length 10mm diameter clear heatshrink tubing [Jaycar Cat WH5555]
- 2 DS18B20 digital temperature sensors in waterproof housings [eg, uk.banggood.com item 1211828]
- 1 10k Ω lug-mount NTC thermistor [Altronics Cat R4112]
- 3 2-pin polarised headers, 2.54mm pitch, with pins [Jaycar Cat HM3402]
- 2 M6 copper crinkle washers
- 2 M6 hex nuts

Don't forget to make provision for some heatshrink tubing for any multi-wire or multi-cable bundles, to keep everything neat when you run them later.

Configuration and testing

It's a good idea to test the unit before making the final connections, because if you find any problems later it will be harder to fix them if the unit is already captive in its case due to wires soldered directly to the board.

You will need to load it with its initial configuration. All you need to do this is a computer with a USB port and a serial terminal program such as Tera Term Pro (a free download from: <https://ttssh2.osdn.jp/index.html.en>).

You also need an internet connection, although it doesn't necessarily need to be available at the same time that the computer is hooked up to the unit; you can prepare the configuration beforehand.

Start by plugging the finished board into your computer using either a Type-A to mini Type-B USB cable (if you fitted CON1) or a chassis-mounting Type-B socket wired into CON3, plus a suitable cable.

Check that your computer has detected a new USB serial device. That verifies that the microcontroller is working correctly. In Windows 10, you can do this by right-clicking on the Start button, choosing 'Settings' from the menu that appears, then clicking on the Devices icon. You should see a device listed with a name like 'USB Serial Port (COM5)'. The COM number will vary.

Open this serial port using your chosen terminal emulator and then type 'status' and press Enter. You should get a status display similar to that shown in Fig.6. If you don't, check your port settings (the baud rate setting and so on are not important).

If you can't get any response, you may have a wiring or hardware fault, so check that your USB socket is soldered and wired correctly, that the PIC chip (IC1) is properly programmed and soldered and that all associated components have been fitted correctly.

Once you've established communications with the chip, open a web browser and go to: <http://bit.ly/pe-dec19-4ch>

This page will help you set up a basic configuration for the unit, for further testing.

See the panel on Settings for help on how to set the unit up initially. The web page above translates your settings into an encoded string which you can send to the *Fan and Pump Controller*, setting its configuration to the desired state. Read up on the basic settings now – you can ignore the more advanced settings for now. You can read about them later, once you've established that everything is working.

Loading the configuration

Once you have selected all the options you want, click the 'Copy to clipboard' button at the bottom of the window, then switch to your terminal program and paste the configuration string (which is now in the system clipboard) into the terminal. You can do this in Tera Term Pro by right-clicking in the terminal window, then pressing Enter.

You should get a response that says 'OK'. If it says 'Error', then the clipboard string has somehow become corrupted.

Immediately after pressing Enter, the new configuration takes effect. Type 'show status' and press Enter and you may see some changes already.

Initial testing

You can now use the 'override' command to perform some basic checks on your settings. The override command lets you ask the unit to pretend that the supply voltage or temperature sensor readings are a particular value, so you can see what happens without actually having to vary the supply



There's not a huge amount of space under the hood of many cars, especially a big V8! Choose a location that doesn't interfere with the operation of any other controls and, preferably, is easy to get to! Ensure all wiring is adequately secured.

voltage or heat up or cool down the sensors. This is useful both when the unit is installed in the vehicle (since you can't always get the sensors to read what you want while idling) but also at this early stage, to avoid the need for variable voltage sources and variable resistors.

First, run the 'status' command (type 'status' and press Enter). Since the unit has no 12V supply, it should give a supply reading close to 0V and it should indicate that it is in sleep mode as a result. Now issue the command 'override supply 14.4V' (or similar). Re-run the status command. You should see that the supposed supply voltage has increased and that the unit is now in run mode.

However, since it knows there is no 12V supply, it will not drive the MOSFETs, to protect the driving circuitry (which runs off the currently non-existent 12V supply).

Still, you can see what PWM duty cycle the unit will drive each output to for the current temperature sensor inputs. You can then issue a command like 'override TS1 47.5C' to make it pretend that temperature sensor #1 is actually at 47.5°C, rather than its actual current temperature. Re-run the status command and observe how the output duty cycle(s) change.

You can then override other sensor temperatures, or change the existing one, to see what happens. If it isn't working as expected, review your configuration and repeat the procedure above to load the new configuration into the unit, then continue testing in this manner. See Fig.6 for an example where the override feature is used.

Once you have finished testing, issue the 'override clear' command and the unit will go back to working as usual. You can then proceed to connect actual loads if you want – they don't have to be fans, a 12V LED would work and would give you an easy way to see how the duty cycle changes.

Having said that, since your fan(s) will already have the right connectors, it may be easiest to use them for testing. Just make sure you have them in a safe location so that when they are powered up, they don't fall over and the spinning blades won't hit anything.

You will also need to connect the sensors (if not already connected) and a 12V power supply with sufficient current capability for further testing. This could be your car battery. You can also use the override command in live testing.

It's also a good idea to check that the sensors are actually working, rather than just relying on the override command. Test each sensor by heating it up or cooling it down slightly, then re-run the status command and check that the temperature reading from that sensor has changed as expected.

You can use a hot air gun, some ice or a cigarette lighter; just make sure if you are heating the sensor that you don't overheat it or anything nearby. For example, if using a lighter, keep the flame some distance below the sensor and don't heat it for more than a few seconds.

You may also be able to observe the fans/pumps being driven, depending on whether you're pushing the sensor temperatures into the ranges where those loads are activated.

Preparing the case

Now you need to figure out where each wire is going to enter the case. Try to keep in mind the layout of the pads and connectors on the PCB; ie, avoid wires crossing all over the place inside the box, if possible. Mark and drill the holes required to get those wires into the case. Don't make the holes any larger than necessary.

Solder the fan/pump and power supply wires onto the pads, in the locations shown on Fig.4. It helps to pull these as far into the box as necessary, so you can do the soldering outside the box, then pull the wires back out when you have finished.

The other connections are made with polarised plugs. Depending on the sizes of the holes you've made, you may be able to crimp/solder these onto the wires and then feed them through the holes, then push them into the plastic plug blocks. If they don't fit through the holes, you will have to feed the wires through first and then crimp/solder the pins afterwards.

Note that the LED and any DS18B20 temperature sensor wires are polarity sensitive, so make sure you refer to Fig.4 and get them on the right side of each plug. The enable/disable and any NTC thermistor wiring is not polarity sensitive, so the pins can go into the plugs either way around.

While it isn't necessary to bring the USB connector outside the case – you could just open up the case and plug in a cable if you need to change the way the unit operates – it's

Explanation of Settings

Basic Settings

The settings user interface (available at: <http://bit.ly/pe-dec19-4ch>) is shown in Fig.5. Note that this has been revised slightly since the December article, to remove some unnecessary features and add some other useful ones. Start by using the top four drop-downs to select the type of temperature sensors you have hooked up to CON4-CON7.

The following three voltage thresholds control how the unit responds to changing battery voltages. The defaults are sensible, so you don't necessarily need to change them.

The first determines the voltage the battery needs to rise above before the unit will become active.

The second determines the voltage it must fall below when active to terminate normal operation and enter cool-down mode, an optional time during which the fans and/or pumps will continue to run, possibly with reduced duty cycles.

The third voltage threshold prevents cool-down mode from flattening the battery. If the battery voltage falls below this during cool-down mode, the unit will immediately go into sleep mode and wait for the battery voltage to rise above the switch-on threshold before becoming active again.

The cool-down delay is designed so that vehicles which charge the batteries sporadically will not enter cool-down straight away when the battery is no longer being charged. The battery voltage must be below the 'Enter cool-down' threshold for this long before it will go into cool-down mode. For vehicles which continuously charge the battery, set this to a short time (eg, 1s).

The minimum cool-down on-time sets the minimum time that the unit must be in full operation before it goes into cool-down mode. If the battery voltage is above the threshold for a shorter time than this, the unit will immediately shut down instead.

The cool-down time is the maximum number of seconds that the unit will spend in cool-down mode before shutting down.

Cool-down compensation allows you to reduce the fan/pump duty cycles in cool-down mode, compared to what they would be during normal operation given the sensor temperatures. Upon entering cool-down mode, the duty cycles are immediately multiplied by the maximum value of this setting. So if that is 75%, they will drop by 25%. The minimum duty cycle setting for each output will still be in effect.

As the battery voltage drops towards the shut-down threshold, the duty cycle multiply value approaches the lower value of the setting. So with the default values, duty cycles will reduce from 75% of nominal to 25% of nominal before the unit shuts off completely.

Per-output settings

Each output has a similar configuration entry in the table beneath the global settings. You can enable or disable each output individually using the drop-downs at left. You can also set output #2 to be a slave to #1 so that the two outputs can be paralleled to give a single 20A output. The same comment applies for outputs #3 and #4.

The PWM frequency must be the same for outputs #1 and #2 and the range of possible frequencies is shown on-screen, along with the closest frequency to the one you have selected, which will be the actual frequency used. Note that the real frequency will also vary slightly depending on the micro's oscillator calibration.

The frequencies for outputs #3 and #4 can be set independently but only if one of them is 10Hz or less. The maximum frequency setting for these two inputs is 2kHz. Typically, you would only use two different frequencies if one of these outputs is controlling a pump and you want it to be driven with long pulses. In this case, you can choose a frequency as low as 1/10Hz (100mHz).

The duty cycle for the output is determined by three main parameters: the duty cycle range, the temperature range and the way the sensor data is combined. The lowest duty cycle in the range given will occur when the sensor reading is at the lowest temperature specified, and the highest duty cycle will occur when the sensor reading is at the highest temperature specified.

In other words, if you set the duty cycle range to 40-60% and the temperature range to 20-30°C, you will get a duty cycle of 40% at 20°C, 42% at 21°C, ... 58% at 29°C and 60% at 30°C.

In the simplest case, this temperature is derived from a single sensor. This is the default; you will find that initially, the duty cycle of output 1 is derived from TS1, of output 2 from TS2 and so on. But you can change this mapping. Multiple outputs can use the same sensor if desired.

The final setting we'll describe here is the ramp rate, which specifies the minimum number of milliseconds that it takes for the output duty cycle to change by 1%. So if you set this to, say, 100ms then a change from 0% to 100% duty cycle will take 10 seconds.

Advanced settings

The Curve setting for each output allows you to compensate for loads where the speed/power is not directly proportional to voltage,

linearising their speed-to-temperature relationship. For example, if you have a fan where speed is proportional to the cube of the average voltage across it, use the Cube Root setting to provide a more linear speed with temperature.

SVC stands for Supply Voltage Compensation, and allows the duty cycle to be automatically dialled back as the battery voltage increases, providing a constant voltage/speed for a given input temperature. Simply specify the voltage at which you want this to take effect (eg, 12V). If the supply voltage is, say, 13V then the duty cycle will be reduced to 12/13 of nominal to give the same average voltage across the load.

Advanced temperature formulas

To the right of the sensor name, you will see a minus sign and then a drop-down box containing zero.

You can select a different number to offset the sensor reading or, more usefully, you can select a second temperature sensor to make a differential reading. The temperature settings you enter for 'Temperature range' then refer to the difference between the two sensors.

Rather than using a single sensor on either side of the minus sign, you can instead change the blank dropdown in front of it to read 'min' or 'max' and this will let you select a second sensor.

The temperature used in the calculation will then be the lowest (min) or highest (max) of the two readings. Or you can make one of the values a constant; the temperature sensor reading will then be clamped when it goes below (min) or above (max) that value. That feature is most useful in the differential sensing mode.

So effectively, you can build a simple formula to derive the temperature reading from up to four sensors, rather than just using the temperature from one sensor directly.

There is one additional option; you can actually have TWO such formulas, using the same structure (but they can be different).

The unit will calculate both values and then the result will be either the lowest (min), highest (max) or average (avg) of the result. That gives you a further way to combine multiple temperature readings.

To enable that option, click on the first black drop-down in the temperature measurement box and change it to one of the three other options. The second formula will then appear, and you can fill it in.

certainly more convenient to have it available from the outside. This is especially true if the unit is going to be buried behind panels or under other bits of the vehicle.

We've provided the option to fit a waterproof USB socket on the outside of the case and connect it via pin header CON3. Simply wire up the USB socket pins as per Fig.4 – the standard USB wire colour codes are shown there too. But

in many cases, it will be easier to feed a micro-B to Type A USB cable through a hole in the box and plug it into CON1 on the board, then seal up the hole with silicone sealant.

Tuck the USB plug away somewhere that it won't get splashed with too much water and tie it up with a twist tie or two so that you can easily remove it and plug it into a laptop later if you need to reconfigure the unit.

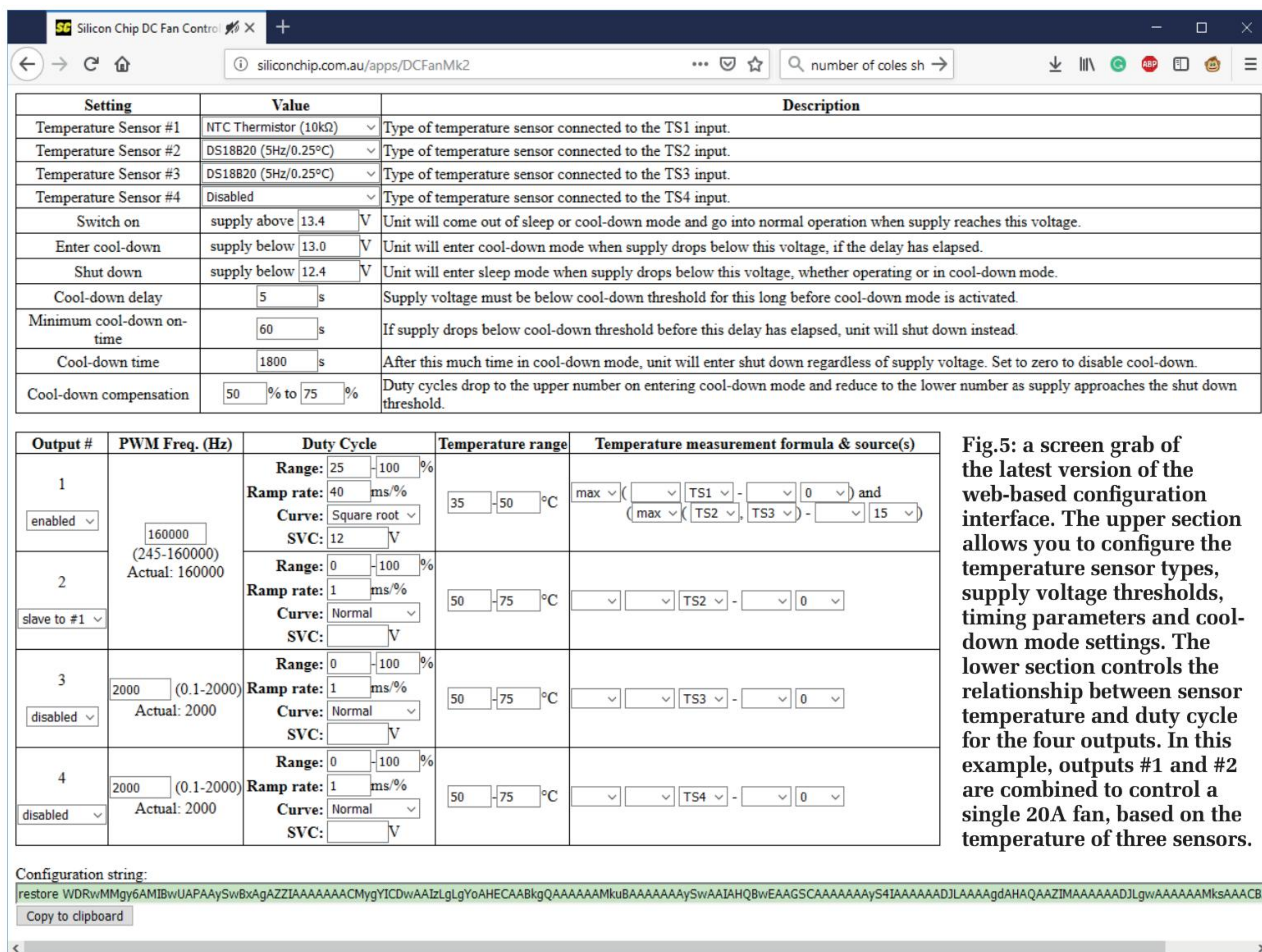


Fig.5: a screen grab of the latest version of the web-based configuration interface. The upper section allows you to configure the temperature sensor types, supply voltage thresholds, timing parameters and cool-down mode settings. The lower section controls the relationship between sensor temperature and duty cycle for the four outputs. In this example, outputs #1 and #2 are combined to control a single 20A fan, based on the temperature of three sensors.

That's the approach we took in our installation

Once you have fed all the wires in through the holes you've made in the box, solder and/or plug them into the board where required. If you're paralleling outputs to get the 20A current rating, you can run a short wire link between the negative pads for the master and slave outputs. There's no need to link the positive pads since they all join to the 12V supply anyway.

It's then time for a final test before locating the PCB neatly in the case and securing it with neutral-cure silicone sealant. Use the same sealant in generous quantities to seal up all the gaps around the wires where they enter the case, on both sides. Even if it looks like a tight fit, hit it with the goop, just in case!

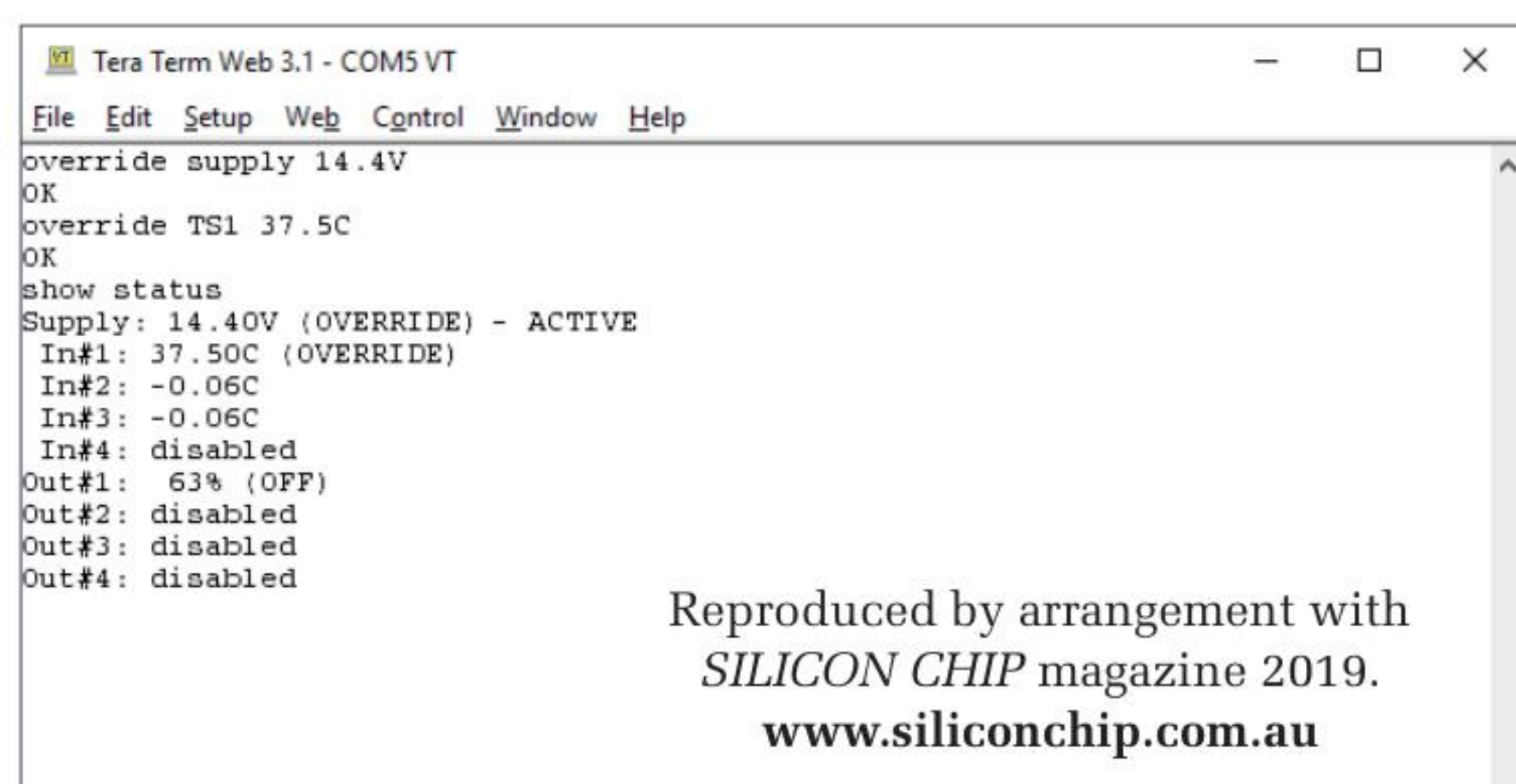


Fig.6: this shows how you can use the override command in the USB serial terminal to test the unit. You can set pretend supply voltages and sensor temperatures and observe how this changes the output duty cycles. If you have fans and a power supply connected, their speeds will change as if the sensor temperatures have changed to the values given.

Make sure none of the wires are too taut either; you want a little slack inside the case to allow for a small amount of movement. Leave it for a few hours to set.

You can then locate it in your vehicle, in the place determined earlier, and tie it down using screws, cable ties or

List of USB serial terminal commands

status – shows the unit's current status, including sensed battery voltage, sleep/cool-down/active state, sensor temperatures, PWM output duty cycles and the override status.

dump – displays the unit's configuration string (including restore command) on the console. This can be pasted into the web app to retrieve the current configuration.

restore – when followed by a base-64-encoded string of the appropriate length, updates the unit's configuration in RAM with the new settings (get this from the web app).

save – saves the current configuration in RAM to Flash, so it is retained the next time power is cycled. Usually used after a restore command.

revert – loads the configuration from Flash into RAM, overwriting any changes which have been made but not saved since power-up.

override supply xx.xxV – pretend that the supply voltage is the specified value until cleared.

override TSn xx.xx°C – pretend that temperature sensor n is at the specified temperature until cleared.

override clear – clear all current overrides.

Common automotive connectors

Deutsch connectors

We have used two different types of waterproof connector on our prototype. For the two DS18B20 sensors, we used a single 4-pin Deutsch plug and socket set (Jaycar Cat PP2149). This was cheaper than two 2-pin plugs and sockets (Jaycar Cat PP2150). A 6-pin version is also available (Cat PP2148).

Deutsch connectors are used widely on vehicles and are known to be reliable, with a typical current rating of 13A/pin. They are relatively easy to put together, although there are a few steps, and ideally, you should use a specialised crimping tool (but you can get away without it). Jaycar sells an appropriate tool, Cat TH2000, which also requires a Deutsch die set (Cat TH2011).

First, if the wires you will be attaching to the connectors are part of a multi-core cable, you will need to strip back about 20mm of the outer insulation to expose enough wire to feed into the connectors.

You need to strip about 3mm of insulation away from the end of each wire to crimp into the pins later.

Both the plug and socket have a thick gasket inserted into the rear, with a small hole for each wire. The first step is to carefully prise this out of each shell and then push wires through these holes. If your wire is particularly thin (as is the case with the waterproof DS18B20 sensors), use heatshrink tubing to make the wire diameter larger so it will seal properly when pushed through.

The next step is to crimp the wires onto the pins. One set has pointed ends and the other set have cups in the end, which accept the pointed ends of the other pins. The cupped pins are larger so you can figure out which shell they go into by checking for the one with the slightly larger holes.

Once you've figured out which pins will go on which wires, fold the larger metal leaves around the wire insulation, crimping them to hold the wire in place. Next, fold the smaller leaves around the exposed copper. A Deutsch crimping tool will do all this in one step but if you don't have one, you can use small pliers (ideally with angled ends) to carefully fold the leaves around the wire and clamp it down hard. It isn't ideal, but it does work.

The trick to doing this is to make sure that you don't just squish the leaves flat, as they will tend to spread out and make the pin too wide. You also need to compress them horizontally, so that the final crimp is compact.

We also like to add a little flux and then solder to the top of the exposed wires to ensure good electrical contact, but that technically

shouldn't be necessary if the wires have been properly crimped (but that's quite tricky to get right if the wire is very thin).

Once all the pins are soldered, push them into the rear of each housing until you hear them click into place. For the cupped pins, you will know they have been pushed home because their ends will be flush with the front of the connector.

For the pointy pins, it can be quite hard to push them in (especially with the gasket in the way), so you may find it easier to push them in part way and then grab them from inside the front of the shell using pliers, and pull them forward until they lock in place.

Now all you need to do is push both gaskets back into the rear of each shell, making sure that they sit flush with the rear of the connector all around the edge, then push the flat orange plastic piece into the end of the socket (ie, the shell with the cupped pins) until it locks into place. This stops the sealing gasket from being pulled off when you withdraw it from the plug later.

The green plastic wedge pushes into the end of the plug and locks in place in a similar manner.

Narva connectors

This is another type of multi-pin waterproof automotive connector, rated at 20A/pin. They are a bit more expensive than a Deutsch connector but have a higher current rating. Jaycar sells these in 2-pin (Cat PP2110), 3-pin (Cat PP2112), 4-pin (Cat PP2114) and 6-pin (Cat PP2116) versions.

We have used two in our set-up; one 2-pin version for the NTC thermistor on the intercooler radiator, mainly because we already had a suitable plug wired to the existing thermistor in the vehicle, and a 4-pin version to connect the unit to the battery.

Its 20A rating is sufficient for our installation as only one fan is being driven, and the four pins mean we can connect both pairs of battery wires in a single plug/socket.

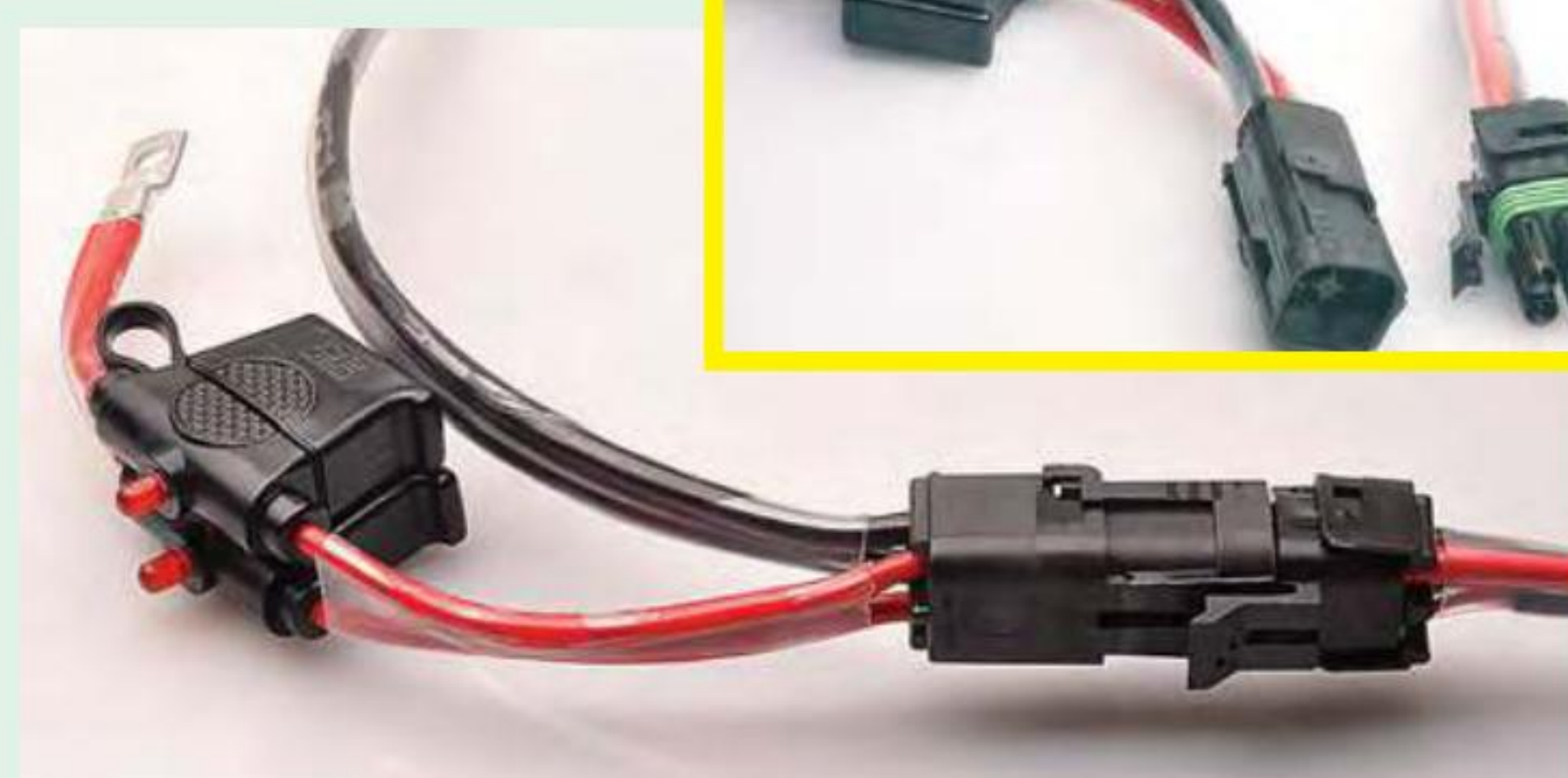
One of the disadvantages of this type of connector is that the socket pins are a bit sloppy and so plugging the two pieces together can be a bit of a chore. But once the pins find the cups, they all lock into place.

Assembling these is similar to the Deutsch connectors but there are some differences. Rather than one large rubber gasket at the rear, there are individual gaskets for each wire, so you need to remember to push these over the wires before crimping the pins (although they can be pushed over the pins if you've forgotten).

Both the plug and the socket have a section at the rear which unclips and swings out, to allow you to insert the pins, which click into place. You then push the gaskets in, leaving the small central section sticking out, then swing the rear back into place and latch it using the plastic clips. This prevents the gaskets from falling out.



Both the Deutsch (left) and Narva (right) connectors are waterproof and are available with various numbers of pins, from 2 to 47(!).



any other method you see fit. As we said, it's a good idea to place some springy foam or rubber between the case and the vehicle to provide some vibration isolation. We used one of the case's two waterproof screw mounting holes to attach it to a cross member in the vehicle.

After another quick check to make sure everything is working, screw the lid on (including the waterproof gasket) and

use cable ties to tidy up the wiring, but again, leave a little slack to allow for movement. Most vehicles flex quite a bit when going over big bumps, especially around the engine.

Tie the USB cable (if using a captive one) somewhere convenient, out of the way but where you can easily reach it once any panels are back in place that you have removed, in case you need to adjust the settings later.

You can tell which is the plug and which is the socket since the socket (which takes the cupped pins) has larger entry holes and is overall deeper.

Note that the gaskets will fit wire rated at around 15-20A. Thinner gauge wire will need to have heatshrink added to form a proper seal while larger gauge wire (~25A) cannot fit through the gaskets (and will only just fit in the connector). You will need to use silicone sealant if you need connectors with heavy duty wiring to be waterproof.

Overall, we suggest that you stick with Deutsch connectors unless your application exceeds their 13A/pin current rating as they are easier to use.

Non-waterproof options

Chances are your fans/pumps will already have a plug and it will be easier if you can find a matching plug rather than cut off the existing one and attach a new one or hard-wire it (although that's certainly feasible).

Our fan already had a '250-series' two-pin connector and these are available from Jaycar too; they sell 2-pin (Cat PP2062), 3-pin (Cat PP2064), 4-pin (Cat PP2066), 6-pin (Cat PP2068) and 8-pin (Cat PP2069) versions.

Make sure you use wire with a high enough current rating to suit your fan. Keep in mind the fan's specified nominal current may be for a 12V supply, and it could draw around 30% more current at 14.4V when the battery is being charged.

Another option for high-current connections, especially to the battery, is Andersen connectors, which are also available from Jaycar. These are available in a range of current ratings including 35A, 50A, 75A, 120A, and 175A. These are dual 'genderless' connectors (ie, two identical connectors will plug into each other).

Individual Anderson connectors are also available, with lower current ratings.

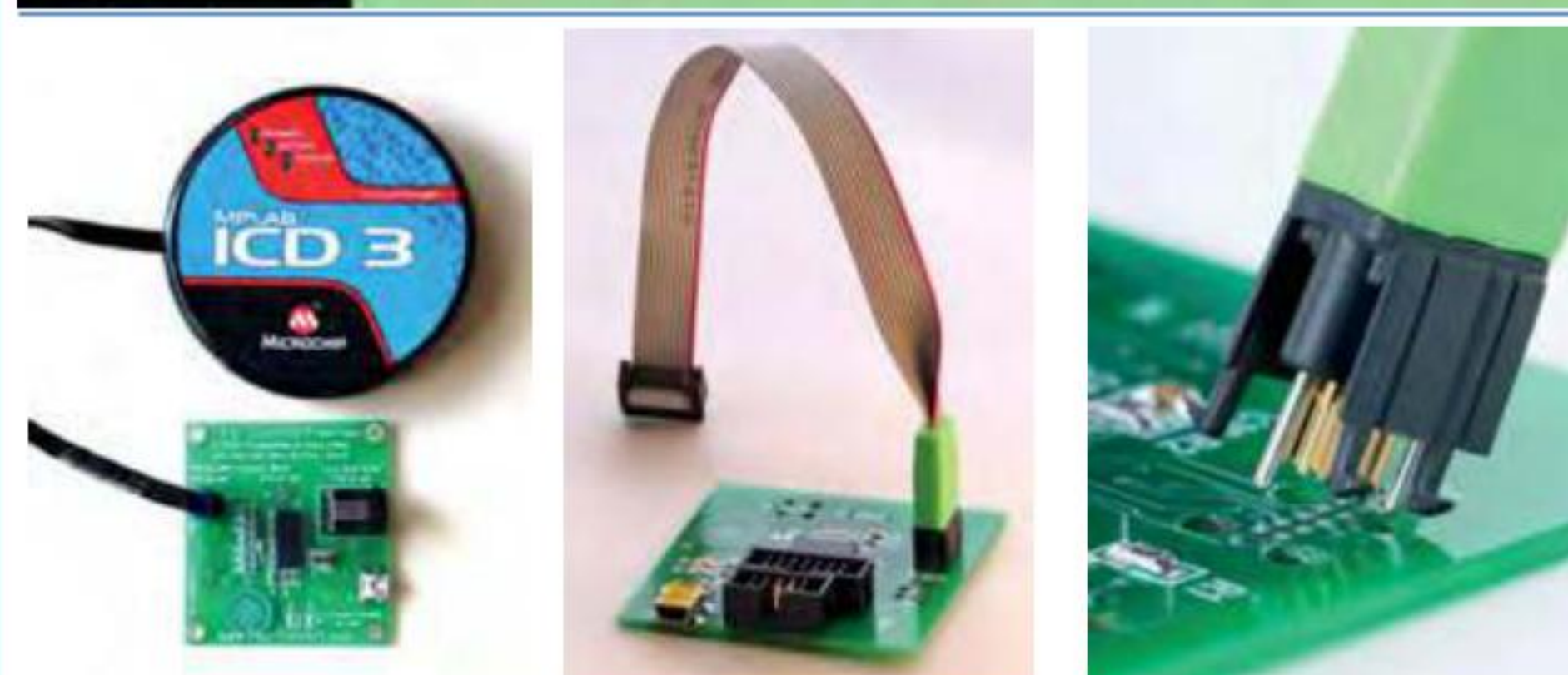
The 50A connectors are quite large but are probably the best choice for battery connections requiring 30-40A. The lower-rated connectors will not accept thick wire and are challenging to assemble, whereas the 50A and up versions feature a 'solder cup' which you can fill with liquid solder and then push the wire into, making them relatively straightforward to put together.

We used the 250-series (right) plug because that's what our radiator had fitted. The two-way Narva connector (below) was used because it had a higher current rating (20A). There are several other types available.



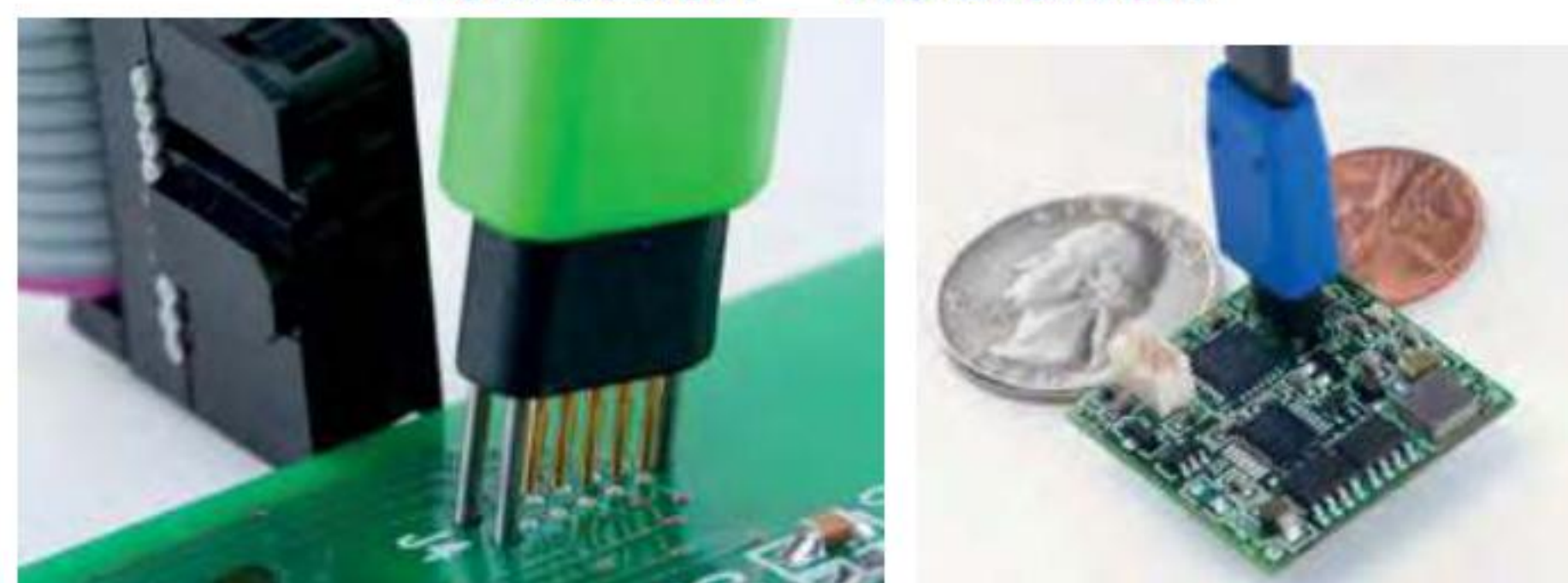
Go for a test drive!

Now all that's left is to go for a drive and make sure everything is working as expected! If you want to leave a laptop plugged in while driving (eg, via a USB extension cable), that's OK, just make sure it's routed in a safe manner (ie, don't leave the bonnet open while driving) and get a passenger to monitor the sensors and fans via the 'status' command.



JTAG Connector Plugs Directly into PCB!!

No Header! No Brainer!



Our patented range of Plug-of-Nails™ spring-pin cables plug directly into a tiny footprint of pads and locating holes in your PCB, eliminating the need for a mating header. **Save Cost & Space on Every PCB!!**

Solutions for: PIC . dsPIC . ARM . MSP430 . Atmel . Generic JTAG . Altera Xilinx . BDM . C2000 . SPY-BI-WIRE . SPI / IIC . Altium Mini-HDMI . & More

www.PlugOfNails.com

Tag-Connector footprints as small as 0.02 sq. inch (0.13 sq cm)

ESR Electronic Components Ltd

All of our stock is RoHS compliant and CE approved. Visit our well stocked shop for all of your requirements or order on-line.

We can help and advise with your enquiry, from design to construction.

3D Printing • Cable • CCTV • Connectors • Components • Enclosures • Fans • Fuses • Hardware • Lamps • LED's • Leads • Loudspeakers • Panel Meters • PCB Production • Power Supplies • Relays • Resistors • Semiconductors • Soldering Irons • Switches • Test Equipment • Transformers and so much more...

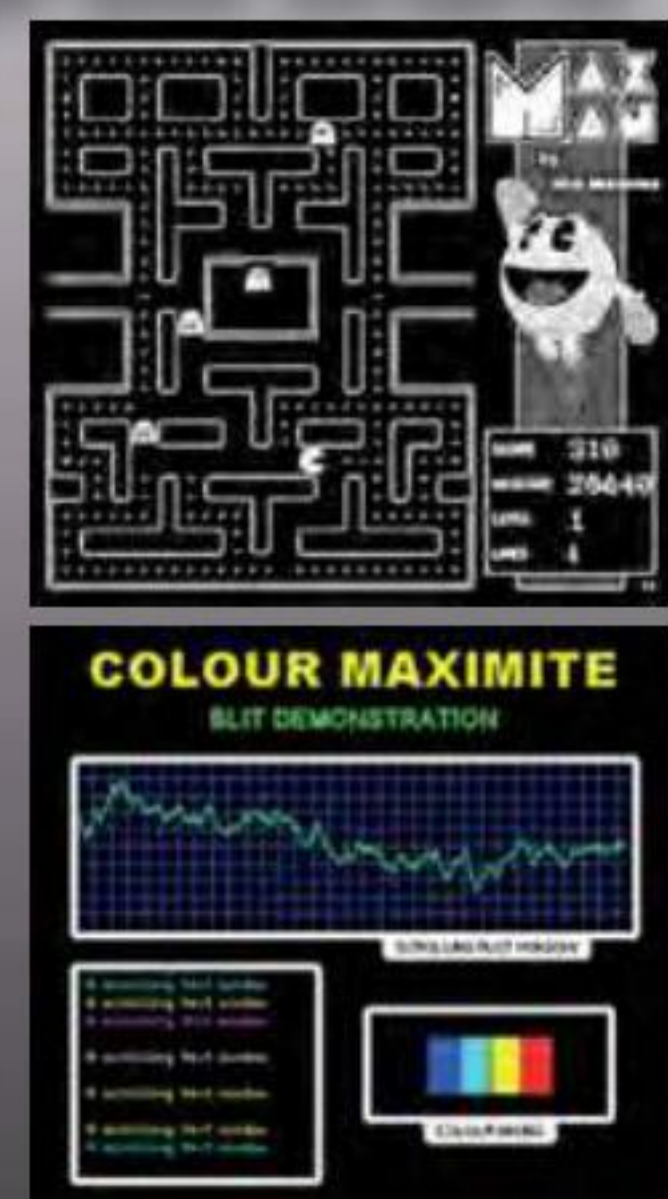
Monday to Friday 08:30 - 17.00, Saturday 08:30 - 15:30



**Station Road
Cullercoats
North Shields
Tyne & Wear
NE30 4PQ**

Tel: 0191 2514363 sales@esr.co.uk www.esr.co.uk

Colour Maximite Computer



Words: Phil Boyce
Design: Geoff Graham

Part 3

A retro 80s home computer with modern-day features

Introduction to using the Colour Maximite Computer with MMBASIC.

Last month, we discussed the assembly of the *Colour Maximite Computer* (or just 'Maximite'). We also covered how to test it to ensure that everything was working as it should, which included running an impressive music demonstration. So if you've been following this project, then by now you should have your very own self-built, fully functioning BASIC computer. As outlined in Part 1, the *Maximite* has a very remarkable specification, and you may have already begun to explore its capabilities. To ensure that you get the most out of it, this month (and next) we will focus on *Maximite* topics and ideas:

- Introduction to MMBASIC (a beginner's guide)
- How to interface hardware (and control it with software)
- Fun with some classic games.

There are so many uses that you can explore with the *Maximite* that it would be impossible to cover all of them here.



Fig.1. Top-left section of the *Maximite* start-up screen (shown when power is applied). The (highlighted) flashing cursor (> _) is the command prompt and indicates that the *Maximite* is waiting for you to enter a command.

However, one thing that will definitely help with a lot of the topics we'll be discussing is the *MMBASIC Language Manual v4.5* (available for download from: <http://bit.ly/pe-nov19-max>).

The *Manual* is packed with over 70 pages of useful information. Naturally, it documents all the MMBASIC commands, and it also discusses some interfacing techniques that will be useful if you need to control external hardware. After you have worked through this month's article, please do read pages 3-6 from the manual; they provide a nice summary of what we're covering here.

Whether you're a complete beginner, software expert, or an electronics genius, we hope you find inspiration in the upcoming topics.

Before we start you will need to connect a few things to your *Maximite* to make it a stand-alone system. These include a VGA monitor, a PS2 keyboard, and (ideally) some speakers. When you power the *Maximite* you should immediately see the start-up screen, as shown in Fig.1. This comprises the *Maximite* logo, the firmware version number, a copyright message, and possibly the time and date (if you fitted the real-time clock option). Underneath all that (and situated on the left-hand side of the screen), there is a small cursor (> _). This flashing cursor can be considered as the command prompt, and it indicates that the *Maximite* is waiting for you to enter a command; so let's get started...

Back to BASICs: a beginner's guide

The *Maximite* can be likened to an 80s home computer, such as a Sinclair, Commodore, Amstrad, BBC Micro, Tandy, Atari and others. These classic machines all used variations of the popular BASIC programming language, yet they all had one thing in common – compared to today's programming environments, they were all extremely easy to use. If you've ever programmed one of the aforementioned computers (probably 30+ years ago!), then I guarantee that when you use the *Maximite*'s version of BASIC (called 'MMBASIC'), it will remind you of those programming days. Let's jump straight in with a well-known example.

Hello World!

At the command prompt, type: PRINT "Hello World!" and then press the Enter key. This will result in Hello World! being displayed. How simple was that?

This is not very exciting, but it demonstrates an important point. The *Maximite* will try to interpret anything that you type at the command prompt immediately after pressing Enter. The command prompt indicates that the *Maximite* is in a particular mode, called **Immediate mode**. When you type in a valid command, on pressing Enter, the valid command is immediately 'interpreted'. Depending on the command, an output (of some kind) may, or may not, be visible. In the above example, the valid command PRINT was entered

with some data ("Hello World!") resulting in the output Hello World! being displayed on the next line. Note the new position of the command prompt, now located on a new line (below the Hello World! output) and once again indicating that the *Maximite* is ready for you to enter another command.

Whenever you type an invalid command (or misspell it) and press Enter, the *Maximite's* attempt to 'interpret' what you meant will fail, resulting in an error message being displayed. Try it (with for example PRITN) and observe the resultant error message. The chances are you will see a syntax error meaning that something was typed incorrectly. Note that the command prompt is now positioned on a new line below the error message and is again waiting for the next instruction.

Here's a more useful example. Type in the mathematical statement `PRINT 15*3 + 27.9/(2*PI)`. On pressing Enter, the result will be displayed on the next line (and the command prompt on the line below that). There are two important points to highlight here:

1. Immediate mode is useful for trying out commands – they are immediately interpreted after pressing Enter.
2. You need to enter a valid command for anything meaningful to happen, so it's worth keeping the *Manual* close at hand to have a good grasp of what commands are available.

MMBASIC's three modes

We have just discussed the Immediate mode, which is useful for immediately testing a command, or for calculating an immediate result. So how is it possible to type in a program into the *Maximite*, when any command(s) entered at the command prompt are acted on as soon as Enter is pressed? Back in the 80s, to write a program a line number was often used at the start of each line. The line number signified that what was about to be typed was actually part of a program (and should not be executed after pressing Enter). The line number would also indicate to the computer exactly where within the program the line of code belonged (in other words, where it needed to be inserted within a program). You could then type the command `LIST` to view your program in a sequence of ascending line numbers. Then, when a program was 'run' (or take your pick from assorted preferred terms: 'started', 'launched', 'activated', 'executed') it would start with the command(s) contained in the line of code with the lowest value line number, and then work through sequentially to the line of code with the highest line number. However, in MMBASIC line numbers are optional (and typically are not actually used at

```
PRINT "Hello World!"
PRINT 15*3 + 27.9/(2*PI)
```

```
ESC:Exit F1:Save F2:Run F3:Find F4:Mark F5:Paste Ln: 3 Col: 1 INS
```

Fig.2. Two sections from the screen: (bottom) MMBASIC's Editor mode is recognised by the yellow line (plus cyan text) at the bottom of the screen; (top) above the yellow line program code is displayed – shown here, our simple two-line program.

all), so – again – how is it possible to enter a program into the *Maximite*? The answer is actually very simple; there is a built-in Editor which is dedicated to entering program code – and when this is launched, you are in the **Editor mode**. Any command entered when in the Editor mode will be ignored (ie, not acted upon straight away when you press Enter) but instead becomes a line of code in the program. The built-in Editor is like a wordprocessor: you enter lines of text (code), with each line of code containing one, or more, commands. If you need to insert some lines of code between two existing lines, then you can simply move the cursor to the required place, press Enter multiple times to insert the required number of 'blank' lines (just like you would do in a wordprocessor) then simply reposition the cursor (with the arrow keys) to the point where you want to insert your code.

Once you have entered the program code, how do you run the program? In the 80s you'd have typed the command `RUN` (and pressed Enter). Well, it is no different on the *Maximite* (if you are in Immediate mode). At the command prompt you simply type the command `RUN` and after pressing Enter your program will execute. At this point, you are in the third mode, called **Run mode**.

When a program is run, the *Maximite* will attempt to interpret one line of code at a time (much as if the line of code had been entered at the command prompt – but without the need to press Enter). Program execution starts with the line of code at the top of the program, and works down through to the last line of code (with the ability to jump to other places in the code depending upon the command used, and possibly the outcome of a certain situation). If, at any point, the *Maximite* fails in its attempt to interpret a command, then the program will automatically stop, switch the *Maximite* to Immediate mode and display an error message on the screen (with the command prompt then shown on the next line down). To summarise, MMBASIC has three modes:

- Immediate mode (testing commands)
- Editor mode (typing in a program)
- Run mode (running the program).

The *Maximite* can only ever be in one of these modes at any moment in time (which makes perfect sense) and you

can easily switch between these modes, typically by pressing a single key, and/or by typing a single command. This is summarised in Fig.3. We will now perform some mode switching.

Switching between MMBASIC's three modes

Now that we understand the *Maximite's* three modes, and what each one is for, let's put things into practice. This is mainly an exercise for beginners, and once mastered, it makes using the *Maximite* much simpler. As well as demonstrating how to switch between the three modes, we will also show you how to:

- Enter a program into the Editor
- Run a program
- Save a program to an SD card
- Load a program from an SD card.

To perform the last two tasks you will need an SD card inserted into the SD socket. Avoid 'cheap' ones; our recommendation is a branded SD card with 8GB or 16GB capacity. (You can use a micro-SD card inserted into a full-size adaptor). Expect to pay between £5 and £12 depending on brand and capacity.

The program we will be using for this exercise will initially be the two lines we typed above when discussing Immediate mode (`PRINT "Hello World!"` and `PRINT 15*3 + 27.9/(2*PI)`). So, assuming you are at the command prompt, let's start by following the four steps below (see Fig.2 and Fig.3).

1. Enter a program into the Editor

This requires us to launch the built-in Editor which requires a switch from Immediate mode to Editor mode. Referring to Fig.3, type `EDIT` to launch the built-in Editor. Note that the Editor mode is indicated by the solid yellow line across the bottom of the screen with a single line of text underneath indicating some key-press shortcuts (Fig.2). Each line of code is entered as required, and the Enter key adds a blank line. The arrow keys can be used to move the flashing cursor to any position within the code. Anything typed will appear at the current cursor position. The backspace (and delete) keys work as normal, as do Page Up, Page Down, Home and End.

So, at the start of the first line type `PRINT "Hello World!"` and then Enter to move to the next line. This is the first line of code in our two-line pro-

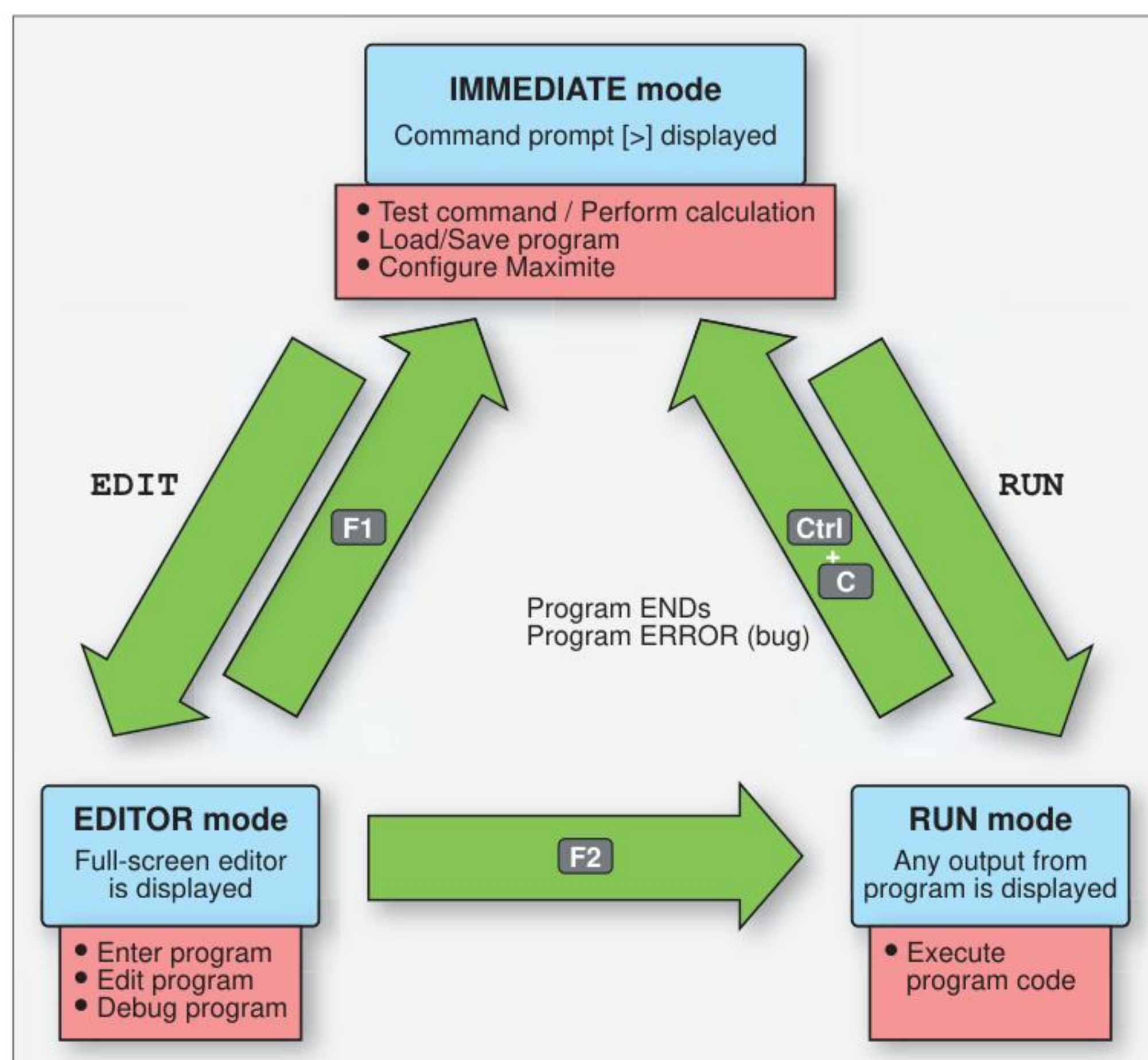


Fig.3. Summary of the three *Maximite* modes and how to switch between them. (Available at the January 2020 page of the *PE* website.)

gram. Then type `PRINT 15*3 + 27.9/(2*PI)` to enter the second line of code and press Enter. In fact, there is no need to press Enter at the end of a line of code for it to become part of the program. This has demonstrated how to enter a simple program; your screen should currently look like the sections shown in Fig.2.

2. Run a program from Editor mode

Look at the key-press shortcuts on the bottom line of the Editor screen, you will see that F2 will RUN the program; ie, switch from Editor mode to Run mode (see Fig.3). Now press F2 and observe what happens. The Editor is closed down, the screen is cleared, and then two lines appear at the top of the screen displaying `Hello World!`, and `49.4404` – in other words, the program was executed. Note that below the two lines at the top of the screen, you can see the command prompt awaiting your next instruction. This is because the program finished running (it reached the end) and automatically returned you back to the command prompt. This is an automatic move from Run mode back to Immediate mode.

3. Run a program from Immediate mode

Having followed the above steps you will be at the command prompt. To run the program from here (as opposed to running it from the built-in Editor) simply type `RUN` (see Fig.3) and your program will run once again. So this is a switch from Immediate mode directly to Run mode. Note that this time the screen does not clear – it simply runs the program from the cursor position.

4. Switching from Editor mode back to Immediate mode

Begin by launching the Editor by typing `EDIT`. Referring to the key-press shortcuts at the bottom of the screen, simply press F1. This will save any changes that may have been made to the code, and then return you to Immediate mode (without running the program). Note that on exiting the Editor, the screen is cleared, and the command prompt is shown on the top line of the screen.

The above has demonstrated how to switch between the three MMBASIC modes shown in Fig.3, and has also shown you how to enter a program into the *Maximite*'s memory. An important point to understand is that the *Maximite* can only have one program in its memory at a time, and if you were to switch off the *Maximite*, the program will be 'lost'. So the next important step we need to understand is how to save

a program in the *Maximite*'s memory onto the SD card, and likewise, how to load a program from the SD card into the *Maximite*'s memory. This is very straightforward as we will now see. By the way, even though the *Maximite* can only ever have one program in its memory at a time, an SD card can contain many hundreds of programs, only limited by the size of programs and the SD card's capacity.

Save and load

First, we will look at how to save a program in the *Maximite*'s memory to the SD card. You do this from Immediate mode by typing the command `SAVE "filename"` (replacing *filename* with something meaningful). Note that with the *Maximite*, the filename ideally needs to be kept to a maximum of eight characters. It can be longer, but will get truncated to eight characters. Now type something like `SAVE "PE_Test1"` followed by Enter, at which point a file is created on the SD card (with the filename specified) which contains an exact copy of the program code in the *Maximite*. Be careful – if you use a filename that already exists on the SD card, then the `SAVE` command will overwrite the 'old' file.

To view a list of filenames (ie, programs) stored on the SD card, use the command `FILES` at the command prompt. Any file listed with a `.BAS` file extension is a BASIC program.

As mentioned above, if we remove power from the *Maximite*, the program in memory will be lost, as we will now demonstrate. Assuming you have saved your two-line program code to the SD card, remove power from the *Maximite*, wait a few seconds, and then reapply power. To confirm this has wiped the program from the *Maximite*'s memory, perform any of the following three simple checks:

1. `RUN` – see that nothing happens
2. `EDIT` – see there are no lines of code
3. `LIST` – see that nothing is listed

Now that we have no program in the *Maximite*'s memory, the next thing to do is to load our program back into the *Maximite* from the SD card. As with most things in MMBASIC, this is straightforward – all you do is type `LOAD "filename"`.

Try it using `LOAD "PE_Test1"` (or whatever filename you specified when saving) and press Enter. To confirm that the program has indeed been loaded back into the *Maximite*, do any of the three tests above (`RUN`, `EDIT` or `LIST`).

Stop a running program

We have now covered all scenarios for typing a program into memory, running the program, saving it to SD, and loading it back into the *Maximite* from the SD card. However, there is one last thing that needs to be covered here, and that is how to stop a running program (and return to the command prompt). The two-line program above was indeed able to reach the end of the program, resulting in the automatic switch back to the command prompt. However, not all programs are like this, and there are three possible reasons why a program may not end:

1. The program has been designed as a continuous program, and is never meant to reach the end; eg, a game
2. A coding error results in a program not responding as it should; for example, an endless loop
3. Not enough time has yet passed for the program to end.

To force a switch from Run mode back to Immediate mode (ie, stop a running program and return to the command prompt) simply press `Ctrl-C`. This notation means that you need to press and hold down the `Ctrl` key, then tap the `C` key and finally release `Ctrl`. To see this in action we will cover one more useful command. To wipe the two-line program currently in memory, we could remove power as we saw above.

However, instead of removing power, we can simply type `NEW` at the command prompt. This removes the program from memory and allows for a new program to be entered. Try it; type `NEW` at the command prompt and check the memory is empty (using any of the three methods from the previous section).

Next, switch to the Editor and type the following four lines of code (a program that never ends unless forced to do so):

```
DO
  PRINT TIMES$
  PAUSE 1000
LOOP
```

When you press `F2` to run the code, you will see the time being updated on the screen (with the *Maximite* remaining continually in Run mode). Now to the point which we are demonstrating here: press `Ctrl-C` and you will see that the program is forced to stop, and return you to Immediate mode.

The exercises so far have provided you with practical demonstrations of everything that is summarised diagrammatically in Fig.3. It is well worth printing this diagram as a useful reference and keeping it on top of your *Maximite*, especially if you are new to MMBASIC.

Interpreted vs compiled

MMBASIC is known as an ‘interpreted language’ because the *Maximite* attempts to interpret each line of code, one at a time, when you run a program (or press `Enter` in Immediate mode). By contrast, in a ‘compiled language’ such as C, you first need to test your complete program code on a computer (eg, a PC) for any syntax errors, then compile the code (convert it into a format that the end computer, such as an Arduino, understands). You then upload the compiled code to the end computer and only then can you run the code. Any program change(s) that you need to make have to be done within the (original) master code on the PC, and then you repeat the whole process of compile, upload, run and test all over again. You need to do this every time you alter the code. True, the time taken to do this can be shortened by using a simulator running on a computer, and a simulator is great if a program is not interfaced with external hardware. However, if it is, then using a simulator becomes much more difficult. So why use compiled languages? Simple – they run code faster. However, they are much more difficult to learn, set up and use. Therefore, a compiled language is not ideal for a beginner because of its complexity and steep learning curve. MMBASIC, on the other hand, is up and running straight away. It’s easy to learn, easy to interface with external hardware and requires no compiling or

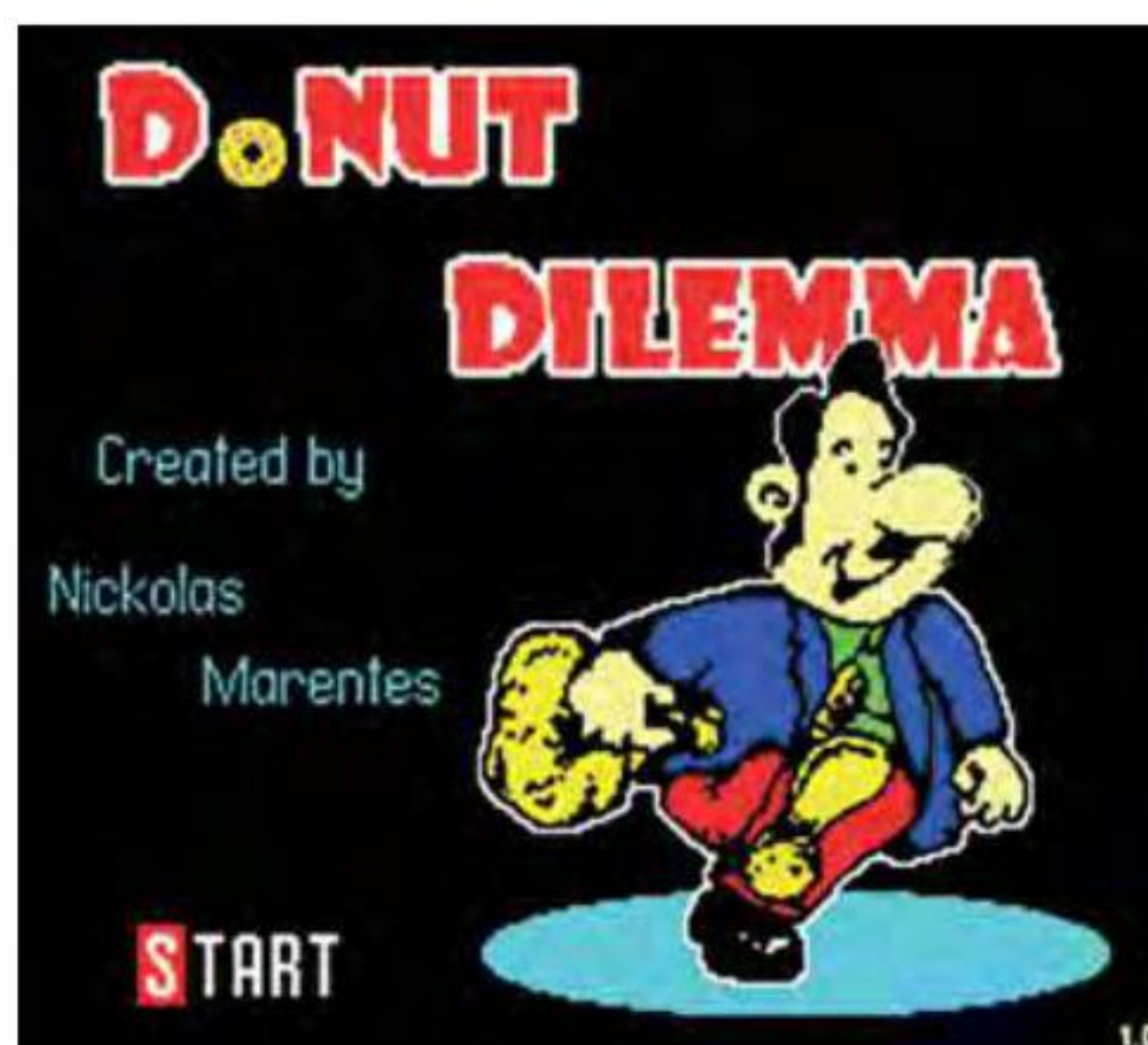


Fig.4. *Donut Dilemma* is a platform game written entirely in MMBASIC.

uploading; beginners can see the results of their code immediately.

Although the big advantage of using a compiled language is that the code runs much quicker than an interpreted language, MMBASIC can run extremely fast on the *Maximite*’s powerful PIC32 microcontroller, and for many applications the time taken to interpret each line of code is effectively insignificant. We will now demonstrate the interpretation speed of an MMBASIC program by running a great game called *Donut Dilemma*, written by the talented Nick Marentes (see Fig.4).

Some Donut fun!

Having covered quite a bit of theory, it is now time to have some fun with a game written entirely in MMBASIC. First though, we have to install the game on your *Maximite*, which will use some of the topics covered above. It will also teach us another new command, `CHDIR`.

You can download the *Donut* program folder from the January 2020 page of the *PE* website. Next, unzip it, and then copy the *Donut* folder to the SD card. Now insert the SD card into the *Maximite*’s SD socket, and at the command prompt, type `FILES`.

In the list you should see: `<DIR> DONUT`. The `<DIR>` indicates a folder, possibly containing other folders and/or files. To switch to the `DONUT` folder we use the change directory command



Fig.6. Your mission in the *Maximite* game, *Donut Dilemma*.



Fig.5. Level 1 in the *Maximite* platform-style game, *Donut Dilemma*.

(`CHDIR`). Now type `CHDIR "DONUT"` and press `Enter`. Next, type `FILES` and you will see an alphabetical contents list of the `DONUT` directory. It includes `DONUT.BAS` (which is the program we will be running to start the game) and also a lot of other files. If you look at the file extensions you will see `.MOD` (sound files), `.BMP` (bitmap images for the backgrounds used in the game) and `.SPR` (a file containing sprite information for things that move).

Now that you are in the `DONUT` directory (and can see the `DONUT.BAS` file), type `RUN "DONUT"` and press `Enter`. The first time you run this program it will quickly copy the sound files to the *Maximite* (a requirement for playing `.MOD` sound files). Once this is done, the game starts.

This is a classic platform-style game (Fig.5). The game’s objective and the keyboard controls are shown in the screenshots of Fig.6 and Fig.7. Remember that this game is written entirely in (interpreted) MMBASIC. For those of you interested, you can take a look at the code (`Ctrl-C` to switch from Run mode to Immediate mode). Have Fun!

Next month

In the final part of this project we will cover some graphical commands, and explain how to use software to control externally connected hardware. We will also load some other demonstration programs and games.



Fig.7. *Donut Dilemma*’s keyboard controls are very typical of 80s games!

PCBWAY

2020 Promotion

Assembly Service Promotion

Only **\$30** for 1-10 pcs assembly

Worldwide **Free Shipping**

10 pcs 1-2 layer PCB only **\$5**

FPC (E-test) only **\$111**

SMT stencil only **\$10**

Get a \$5.00 Welcome Bonus*

DID YOU KNOW WE OFFER:

PCB Prototype To Scale

SMD-Stencils

PCB Assembly

IC Programming

PCB Design & Layout

Multilayer, HDI, High-TG, Rogers, Thick Copper and Halogen-free

Advanced PCB

Aluminium PCB

Flexible PCBs

Components Sourcing

Functional Testing

WHY CHOOSE PCBWay

We produce and assemble prototype and small batch boards in our specialist factories, three major PCB and two assembly production bases in China. We host hundreds of global customer visits and audits every year.

- Best On-Time Shipping Record
- PCB Design Service
- Same Day & Weekend Turnarounds
- No Minimum Order
- Fab + Assembly as Quick as One Day
- 30-day Return Policy

Data-based quality **PCB Prototype** service.



Global Market

4.7%



Daily process orders

2100+



Sponsored projects

42000+



Specialists

70+



Successful Payment

128000+



On-time delivery rate

98.3%



Monthly fabrication

60000+^{mt}



Factories

5+N

Results from Aliexpress Store Feedback & PCBWay Feedback, based on 8,954 + 45,131 Feedback Received.



WOULD REPEAT
YOUR PURCHASE*



POSITIVE FEEDBACK
(PAST 6 MONTHS)*



SHIPPING SPEED
AND DELIVERY TIME*



Immersion gold



8 layer BGA& fingers



4 layers pcb



HASL lead free



20Z Finished Copper



Purple solder mask



Rigid-Flex PCB



Non-framework Stencil



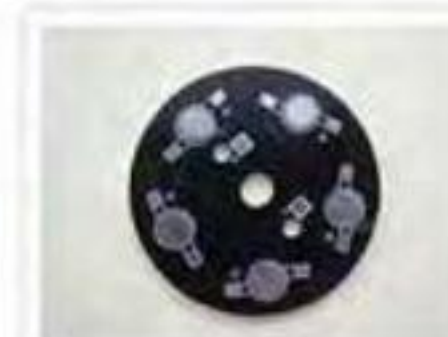
framework SMD-Stencil



Aluminum Board



Flexible PCB



Aluminum Board

PIC n' Mix

Mike Hibbett's column for PIC project enlightenment and related topics

Small, cheap and powerful – Part 5

We continue from the last (November 2019) *PIC n' Mix* by creating a useful application with our tiny PIC10F processor circuit – a temperature and humidity sensor. Periodically reading information from a sensor and writing the values to our LCD, it will make an informative environmental monitor. We will keep the design simple, hopefully running from four alkaline batteries, to minimise additional components and make it easy to construct a self-contained 'product'.

There is a wide choice of temperature and humidity sensors available on the market, but we have some design constraints that help us focus our search. These constraints are:

- Low cost
- Low power
- 2V – 6.5V supply range
- Easy to wire up
- Easy to obtain

DHT12 sensor

We selected the DHT12 sensor, as shown in Fig.1. The sensor itself is the blue box; we chose to purchase it mounted on a tiny prototyping PCB from an eBay supplier. Even when fitted on a prototyping PCB the sensor is small enough to fit into a small enclosure – the sensor board measures just 20×14mm. You can find this particular version on eBay; for example, eBay item 192300097559, but many similar versions are available. The breakout board provides a pull-up resistor and decoupling capacitor on the underside, which are not strictly necessary in this application, so feel free to purchase the sensor by itself.

The circuit for adding the connection from our design to the DHT12 sensor is a very minor upgrade to the

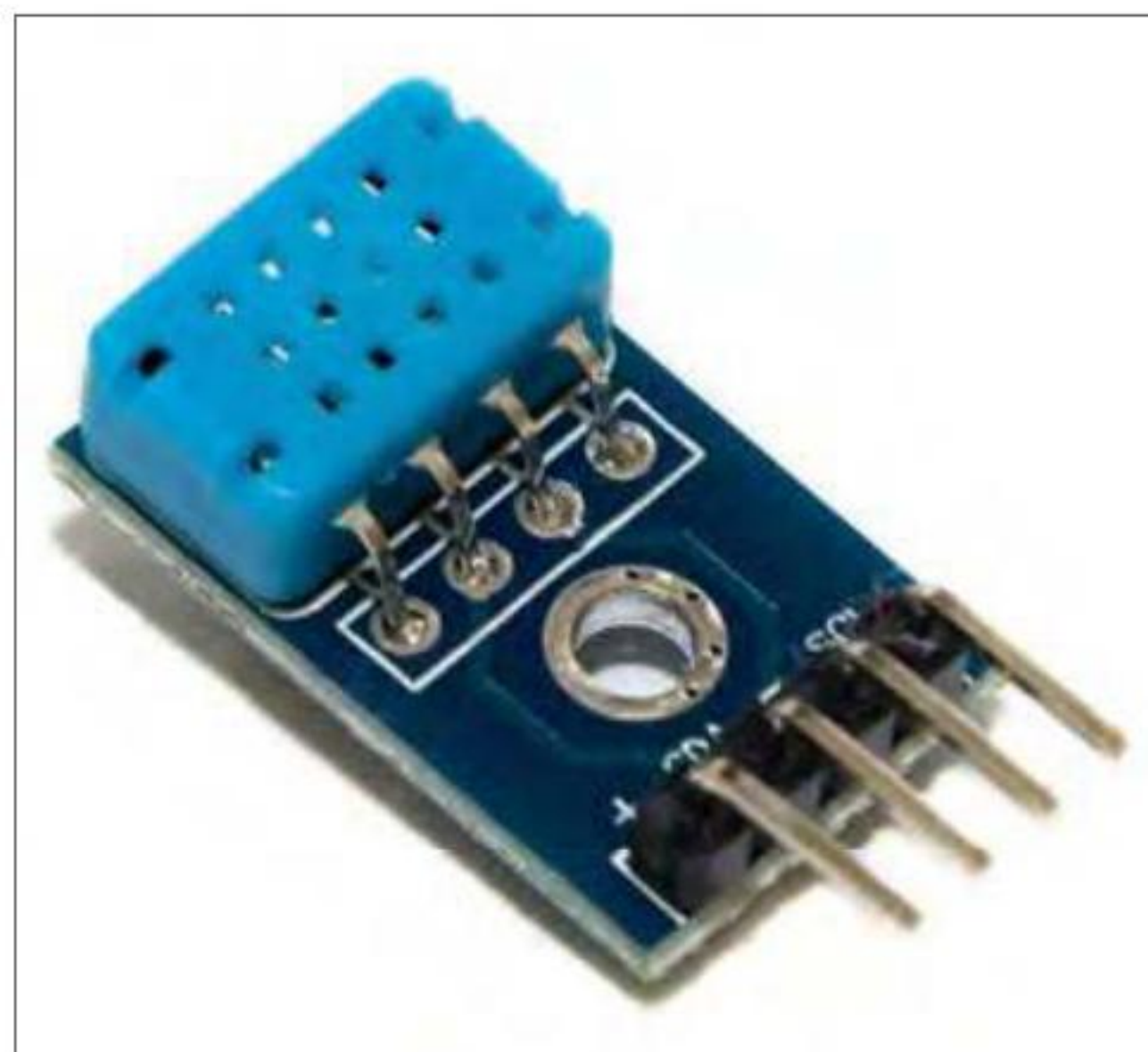


Fig.1. DHT12 Temperature & Humidity sensor (the blue vented component) – you can use the PCB-mounted module version shown here, or the standalone sensor.

November schematic, as shown in Fig.2. We have added just one new connector (top right). On our breadboard, this is no harder than pushing in four wires (we used short male to female patch cables to make the connection. These are incredibly cheap on eBay and well worth stocking up on.)

The DHT12 is a combined temperature and humidity sensor, read via an I²C bus interface. The supply voltage range is stated as 2.7V – 5V, so it will not work across the full range of a four-battery power source, but with low-cost batteries this is not a great concern and a later upgrade could include a simple voltage regulator and additional battery. The current consumption of the sensor is less than 1mA, so power consumption is also not an issue. Furthermore, that is a datasheet value; we expect real-life current consumption to be lower.

Accessing the sensor data is simple: send three bytes to set the device to the start of the sensor data memory, then issue four reads. With our `i2c` functions already implemented in the November article, the additional code to read the data added just a few tens of bytes of code memory through statements like this:

```
i2c_start();
i2c_tx(DHT12_ADDRESS);
i2c_tx(0x0);
i2c_stop();

i2c_start();
i2c_tx(DHT12_ADDRESS + 0x01);

val1 = i2c_rx(ACK_SIGNAL);
val2 = i2c_rx(ACK_SIGNAL);
val3 = i2c_rx(ACK_SIGNAL);
val4 = i2c_rx(NACK_SIGNAL);
i2c_stop();
```

That's all it takes to read the temperature and humidity from the device; no other initial device setup is required. Each time you run that sequence of instructions you will get the latest set of readings. It really is that simple. However, the fun begins when we try to display these values on the LCD.

Heading back to the datasheet, we see that the DHT12 sensor will return values in the format 'nn.y', with 'nn' in the first byte read, and the fractional part 'y' in a second byte read. The values returned are in binary, not the ASCII printable characters required by the LCD.

Temperature values can range between –20 and +60C, and humidity between 20 and 95%. To provide an indication of negative temperatures, the most significant bit of the first temperature byte will be set for negative values. So, how do we manage this data?

Displaying characters

Converting binary values into ASCII printable characters is no challenge for larger processors. On those systems we can use the 'C' programming language library function `sprintf` to do this for us, in a single line of code:

```
sprintf(str, "Temp = %2d.%1d",
val1, val2);
```

Unfortunately, the `sprintf` library function is huge – ranging from several hundred bytes of code space up to several

PIC n' Mix files

The programming files discussed in this article are available for download from the *PE* website.

tens of kilobytes, depending on the processor you are using. It's a very powerful function, but unavailable to us on the PIC10F202. We will have to go back to basics and convert the number manually to a format accepted by the LCD.

What algorithm shall we use? One of the simplest is to perform a divide by 10 to get the most significant digit of 'nn', and then use the mod operator to get the least significant digit. Once again we hit an issue – the PIC10F processor does not have a divide instruction; division by 10 requires a function call by the compiler to yet another large subroutine (divide by 2 or powers of 2 is easy; it's a single shift right instruction that all processors provide. Powers of 10, on the other hand, are not so easy.)

The solution was inspired by a flashback to an old *EPE* article, the *Speed Camera Detector* project from 2005. That project used a special form of data storage called BCD (binary-coded decimal). If we convert the data into BCD, it's a simple trick to convert that data into ASCII, one nibble at a time. By luck, with just two decimal digits to handle, converting the 'nn' digits into BCD is a simple `while` loop in the 'C' language:

```
unsigned char tmp = 0;
while(num > 9) {
    tmp += 0x10;
    num -= 10;
}
return tmp + num;
```

Converting each nibble of the resulting value into ASCII is no harder than adding the ASCII value '0' to it – all possible with the PIC10F instruction set directly. The final code for converting one 'nn' value looks like this:

```
val = i2c_rx(ACK_SIGNAL);
val = DecToBCD(val);
h1 = (val >> 4) + '0';
h2 = (val & 0x0F) + '0';
```

Performing this on all the values read from the sensor takes a few dozen bytes of code memory.

Next, we need to think about the 'main' operation of the device. It's reasonable to code this as a simple loop:

```
LCD_show_banner();
do {
    read_DHT12();
    LCD_show_values
} while(1);
```

This first sets up the main LCD display, then a loop that never ends reads the data from the sensor, displays it on the LCD display at the appropriate

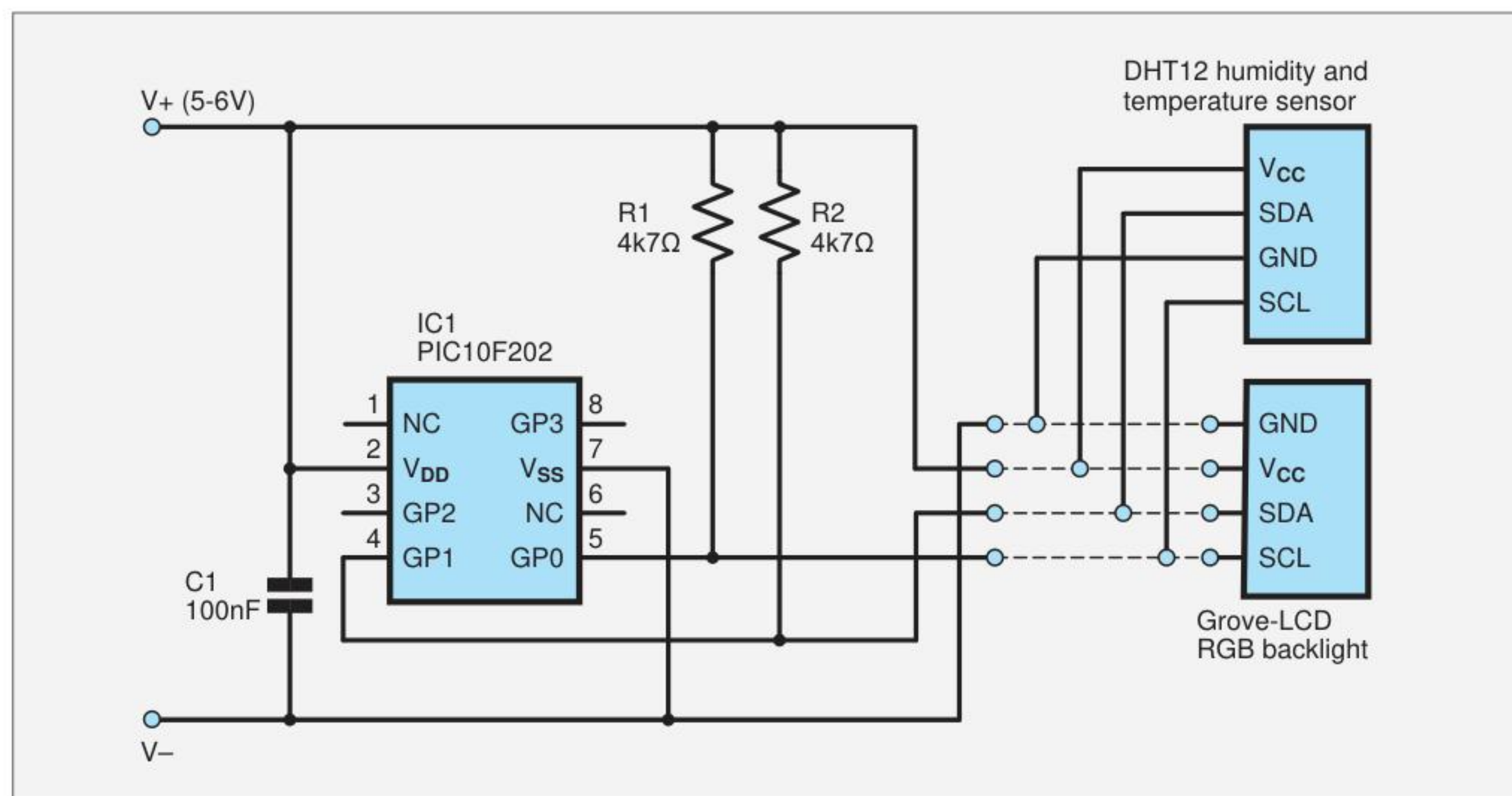


Fig.2. Schematic of the complete PIC10F202 / DHT12 circuit.

positions and loops back to read again. That section of code adds just 20 bytes of code space.

You can look in the source code to see the detail of the three functions mentioned above; the code is available for download from the January 2020 page of the *PE* website.

One interesting point that may raise an eyebrow with experienced software developers is the use of four global variables: `h1`, `h2`, `t1`, `t2`. These variables are referenced in several functions, a technique going against modern software development practices. Their use here is justified due to the simplicity of the application, and limited code space. Better design practices are less efficient; that's not an issue with larger processors, but for small processors like the PIC10F202, we must focus on efficiency rather than modern design practices!

Our application now uses all but three bytes of program memory, and three bytes of data memory. It's reasonable to say that the chip is full. This equated to 410 lines of 'C' source code (including the empty lines.) Not bad for a device with just 512 bytes of program memory.

Display debug

During testing we did see examples where the lack of robustness in our I²C driver routines caused display errors – returning to the lab from an overnight test, the top line of the display had become corrupted, showing some sensor values. This was most likely caused by the LCD command 'move to the second line' being misinterpreted. This issue was easily fixed (or rather, worked around) by including the initialisation of the whole display within the main loop of the program. This way, if the display becomes corrupted, it will be correctly re-written a few seconds later. With this bug corrected in software, we now have just one byte of Flash memory remaining. This device *really* is full now!

It's interesting to note that these errors occurred on our development hardware, which had long cables connecting the I²C bus to the LCD and sensor. These issues may simply disappear when the circuit is moved to a proper PCB with short cables. This brings us nicely on to the subject of designing a proper PCB for this little device.

Professional PCB

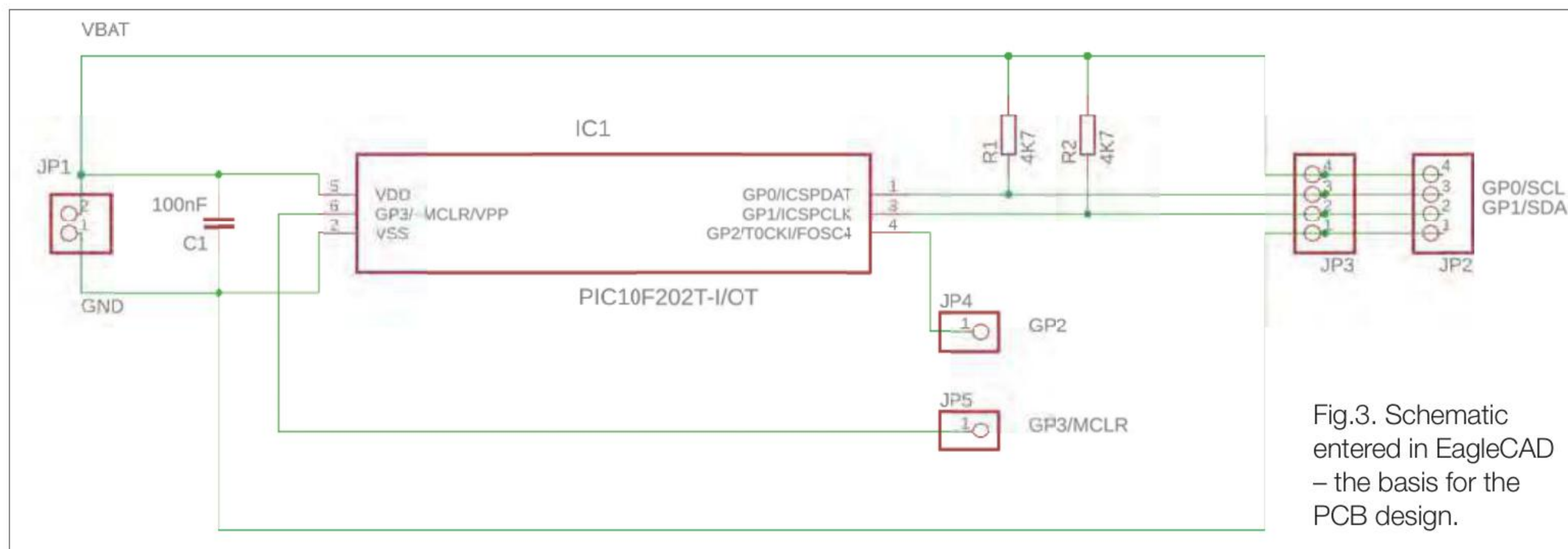
So far, we have been using a breadboard to design and test the circuit; now it's working and providing useful data, it's time to move the design to a PCB.

In the past we would have etched a single-sided PCB (we still have ferric chloride in the lab) but these days the cost of getting professional PCBs manufactured quickly has dropped to a level where it is just not worth bothering. Our favourite supplier is PCBWay in China. For just £21 they will build you 10 PCBs (each measuring up to 100×80mm) and deliver them within five days of order placement. At this price they offer an exceptional range of features – double-sided PCBs, silk screen on both sides and even unusually shaped PCBs.

Our design is going to be very simple and will not showcase the range of services provided, but we cannot recommend PCBWay strongly enough, having used them on dozens of PCB designs. That said, there are other companies offering similar services – we have simply found PCBWay's PCBs to be excellent, and happily chosen to stay with them.

PCB design

To translate the design shown in Fig.2 we will be using EagleCAD. We've covered the use of EagleCAD in articles before, so we won't go into detail of its use, but for us it remains our favourite PCB design tool. For Hobbyist use it's free of charge; for professional use (such as writing this article) it's very reasonably priced.



The first task is to translate the circuit diagram into a format understood by the CAD program. We started with an original drawn on paper, which was then skilfully crafted into a neat schematic by our hardworking team of production elves. That publication version is shown in Fig.2; the design drawn in EagleCAD is shown in Fig.3.

Fig.3 is far less visually appealing than Fig.2, but is now in a format understood by the CAD program. In this design we have added two new test points, GP2 and GP3. This is to allow programming of the PIC10F when fitted to the PCB. Remember, this is a PCB, not a breadboard that we can add wires to easily. These test pads will allow us to easily solder wires to the board and connect those wires to the PIC kit3 to program the device.

Switching to the PCB layout, we first position the components close to each other and reduce the outline of the PCB until we get it as small as possible. The final component positioning that we were happy with is shown in Fig.4.

Hopefully, you have noticed something unusual – these components look nothing like the components we have used before. For the first time in *PIC n' Mix* history, we are creating a project with SMDs (surface-mount devices.)

For hobbyists, SMDs offer one main benefit – being able to create highly miniaturised boards. As an introduction to this aspect of electronics we have chosen to make use of some of the more user-friendly-sized components. Surface mount components can be obtained in significantly smaller sizes than what we have used here.

To get a feel of the scale of these components, Fig.5 shows the four components we have used – both in the breadboard version and for the SMD PCB version. Soldering these components by hand is not difficult, if you have good eyesight or glasses. Magnifying glasses are a good investment, and we will cover this in a separate article later.

With the components positioned where we want them, it's time to add the tracks connecting them – wiring up as

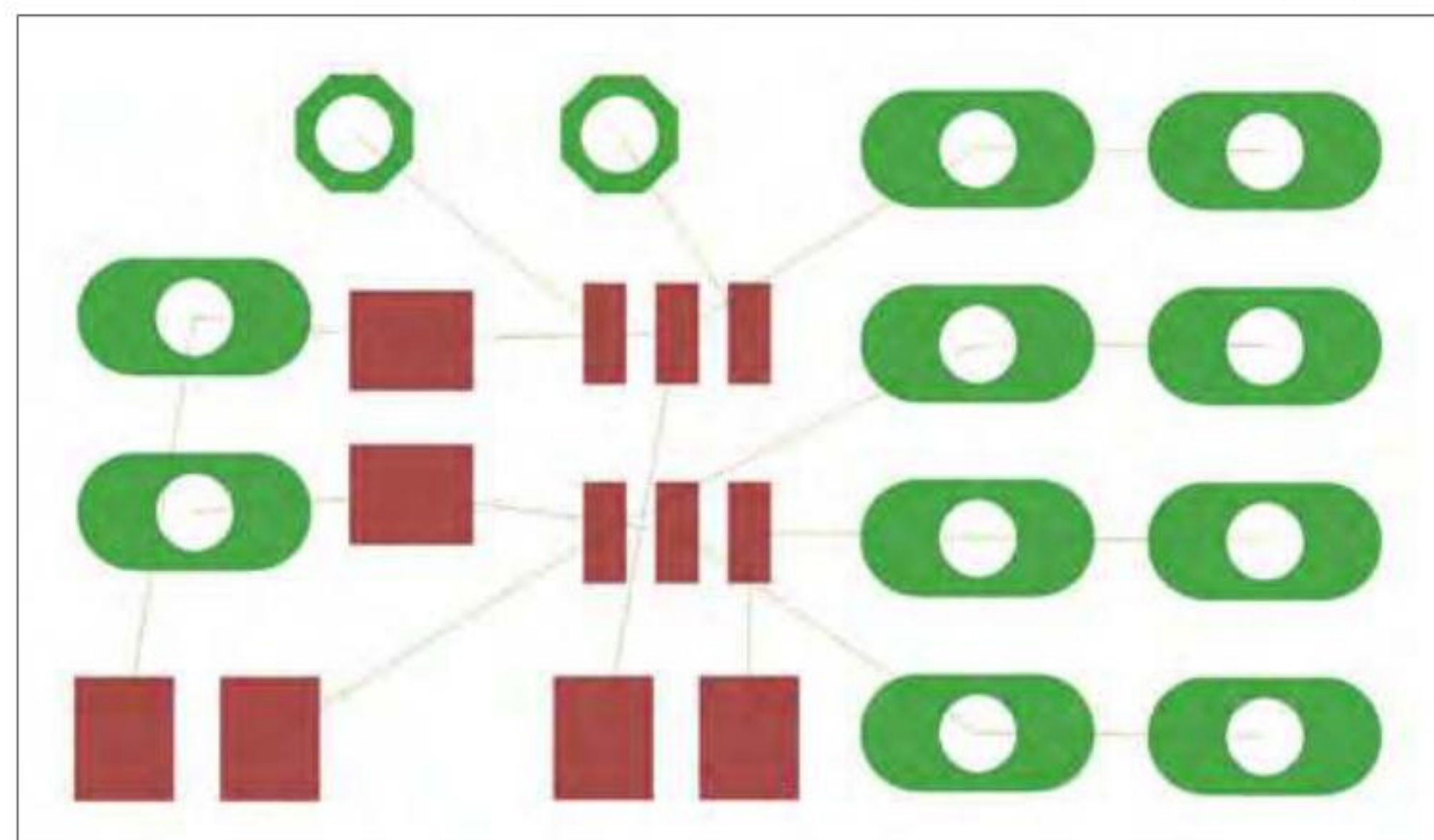


Fig.4. Board layout in EagleCAD – note the fine line 'air wires'.

shown in Fig.3. If you look closely at Fig.4 you can see thin traces known as 'air wires': these show the logical connections between components and it is your responsibility to join these by tracks, on the top or bottom layer of the PCB. Our final PCB design is shown in Fig.6. This image shows track on the top surface in the colour red, and bottom layer tracks in blue. The green colour is for connector pads. This is a simple PCB, so it took just an hour to create.

All PCB design tools store data in their own proprietary format, but for creating printed circuit boards, all tools will provide an export function to save a copy of the design in the industry-standard file format known as 'Gerber'. Within an hour of creating this design we had submitted it to PCBWay and had been notified that the design had passed their internal review process (they check your design for errors such as tracks too close to each other – very helpful, it's caught mistakes in our own designs more than once). After paying £21 securely on-line, we could expect 10 PCBs delivered within five days.

The resulting board for our design is shown in Fig.7, both unassembled and assembled. It's just four components, but they are small.

PCB Assembly

We chose to use surface-mount components for this project, because with so few parts, it makes a great introduction to the techniques involved. We've discussed surface-mount assembly before (see *PE*, February 2019) so we won't go into detail again, but the key points are, use the following:

- Fine solder, 0.4 – 0.3mm diameter
- Fine-nosed metal tweezers
- Fine soldering iron chisel tip.

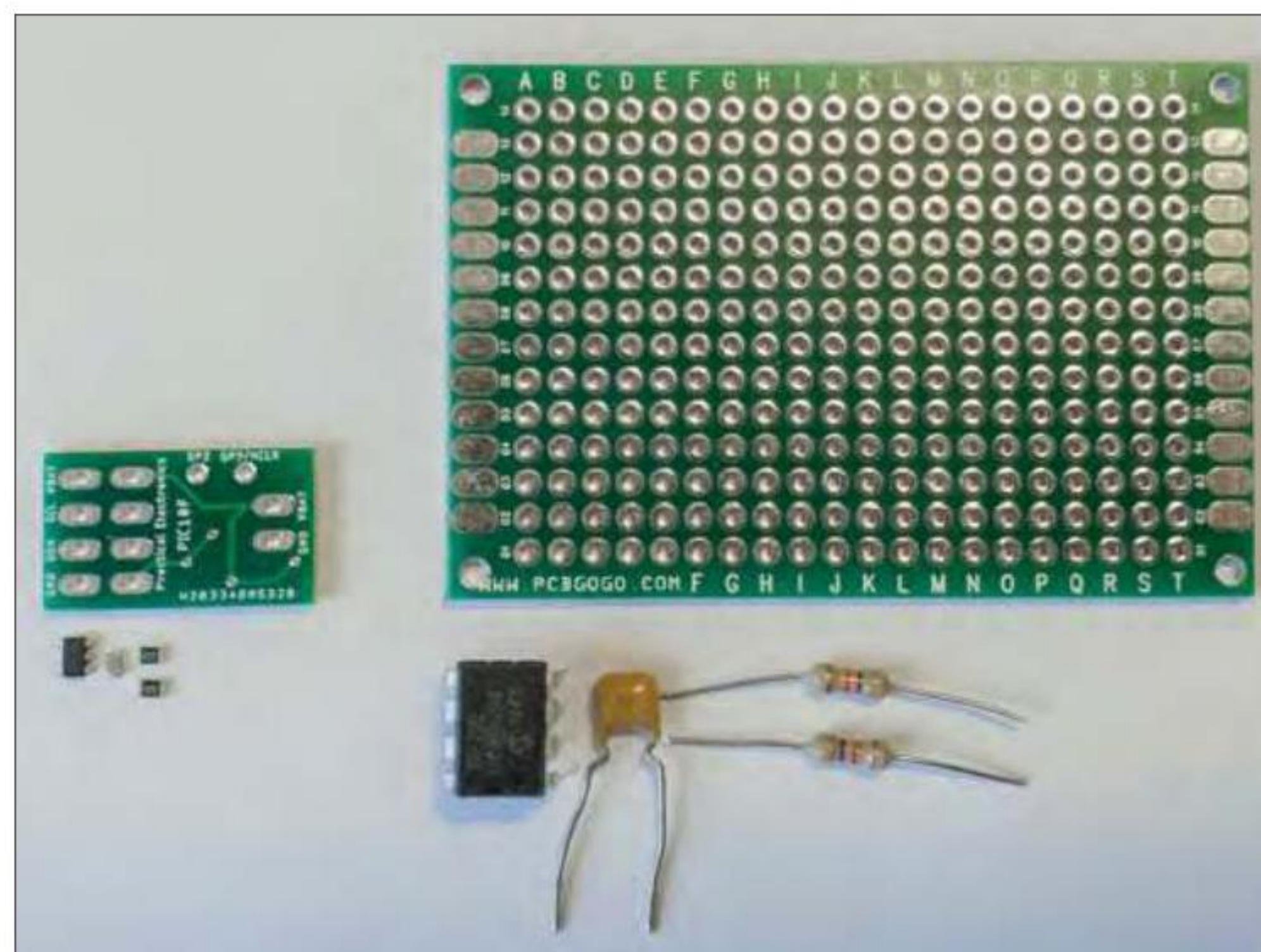


Fig.5. (left) Surface-mount and (right) wire-ended components.

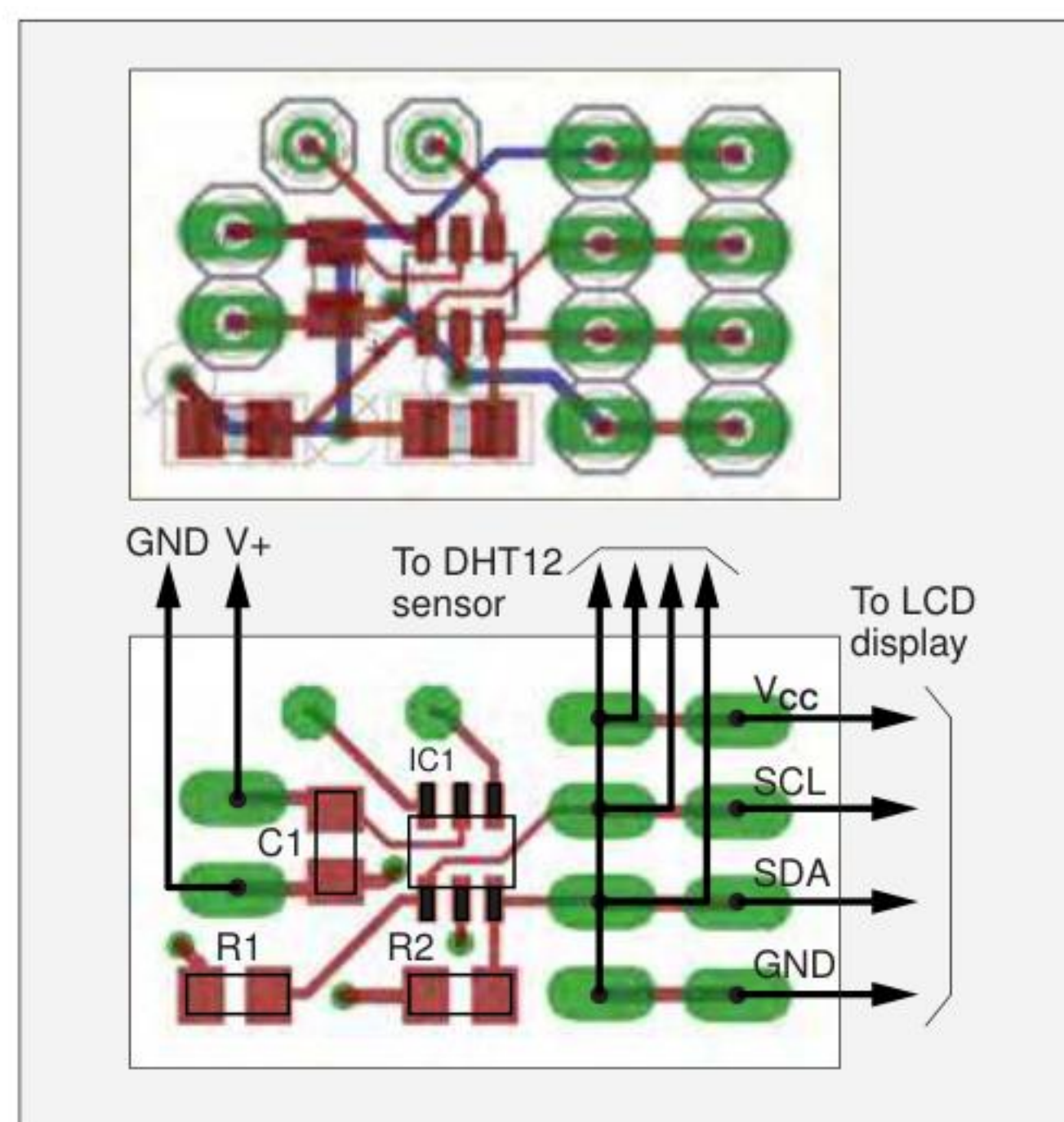


Fig.6. The PCB is 20×14mm, shown here approximately ×2 scale. Top version is the complete EagleCAD view with EagleCAD annotation and both layers (red tracks, component side; blue tracks reverse side; and green for component pads.) This can become quite cluttered, so the lower view shows just the component side with added easier-to-read component annotation.

For the soldering iron tip, a 1mm chisel tip works well. Avoid pointed tips, as these do not transfer heat to the PCB very well.

When assembling the components, first stick your PCB down to your work surface with some doubled sided adhesive tape. This PCB is tiny and will not be easy to hold in place – especially as you will need both hands to solder. Next, apply solder to one of the corner pads of the IC. Pick up the PIC10F IC with your tweezers, making sure the IC is in the correct orientation. Align the IC on top of the pads and solder the corner pin. Now check that the other pins sit nicely over their pads. If not, re-heat the soldered pad, and move the IC around. This can take several attempts.

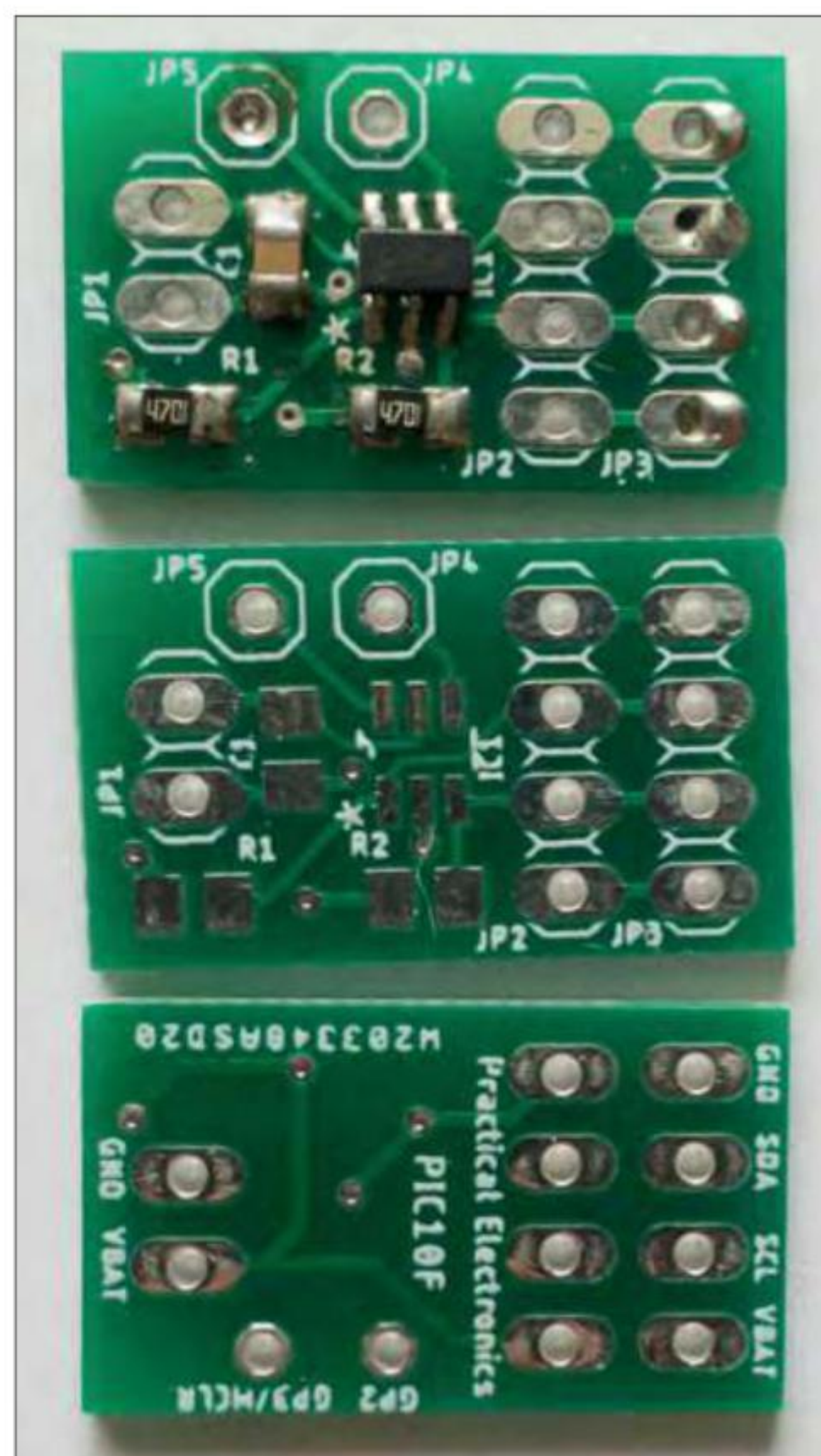


Fig.7. Manufactured PCB: assembled and bare (both sides).

When you're happy with the IC's alignment, solder the remaining five pads. The capacitors and resistors follow the same process but are easier because they have larger pads and there are just two on each component.

Once completed, five wires can be temporarily soldered to the pads for programming before soldering the LCD, sensor and power wires. Our final assembly is shown in Fig.8.

In use

Powering the assembly from a bench power supply we found the device works best at a supply voltage of 4.5V – 6.0V, drawing just 1mA with the backlight turned off, and 10mA with it turned on. A voltage of 5.0V gave the best display contrast, so we powered the assembly from a smartphone USB power supply.

The critical part in this design is the LCD; it's quite sensitive to the voltage range powering it, and is not suited for direct battery supply without a small DC-DC converter (which can be bought on eBay; eg, eBay part 193048101866, for around 99p.)

The full software and PCB design files (EagleCAD and Gerber format) are available for download from the January 2020 page of the *PE* website. Alternatively, PCBWay offer a service to purchase PCBs directly, without requiring that you supply the Gerber files. See the link to the **PCBway.com** website at the end of the article.

Conclusion

At the end of this series of articles we have created a useful product with the PIC10F202 processor and tested the limits of the device. We have also learned some useful lessons:

- Unless your code is very simple, debugging any issues without a proper debugger set-up is challenging
- Choose your sensors carefully. Make sure the output data these sensors provide can be decoded efficiently. Look at the output data formats of the devices you are considering; some may be more friendly to low-resource processors than others
- Given the difficulty of solving issues without debugging capabilities, is this microcontroller actually worth the extra trouble?

For that final question, the answer is, 'it depends'. If you have the space for a larger microcontroller that provides on-chip debugging via a PIC kit 3/4 debugger, then unless your application is extremely simple, you should use the larger device. Fault finding software issues without a debugger is hard. It has, however, been fun designing a fairly complex project that fits in a tiny space, and solving those hard problems was rewarding – and this is why, as hobbyists, we do these things. The knowledge gained is invaluable. By coincidence, at work a production requirement come up using a DHT12 sensor. 'Has anyone used a DHT12 before?' Guess who grinned!

Next time

In the next article we return to the *Audio Spectrum Analyser* project (see *PE*, March 2019), advancing the design with bug fixes and new capabilities.

Links

PIC10F202 data sheet
<http://bit.ly/pe-jan20-pic10f>

DHT12 data sheet
<http://bit.ly/pe-jan20-dht12>

PCBway – *PE*'s recommended PCB manufacturer
PCBway.com

PCBway project link
<http://bit.ly/pe-jan20-pcbw>

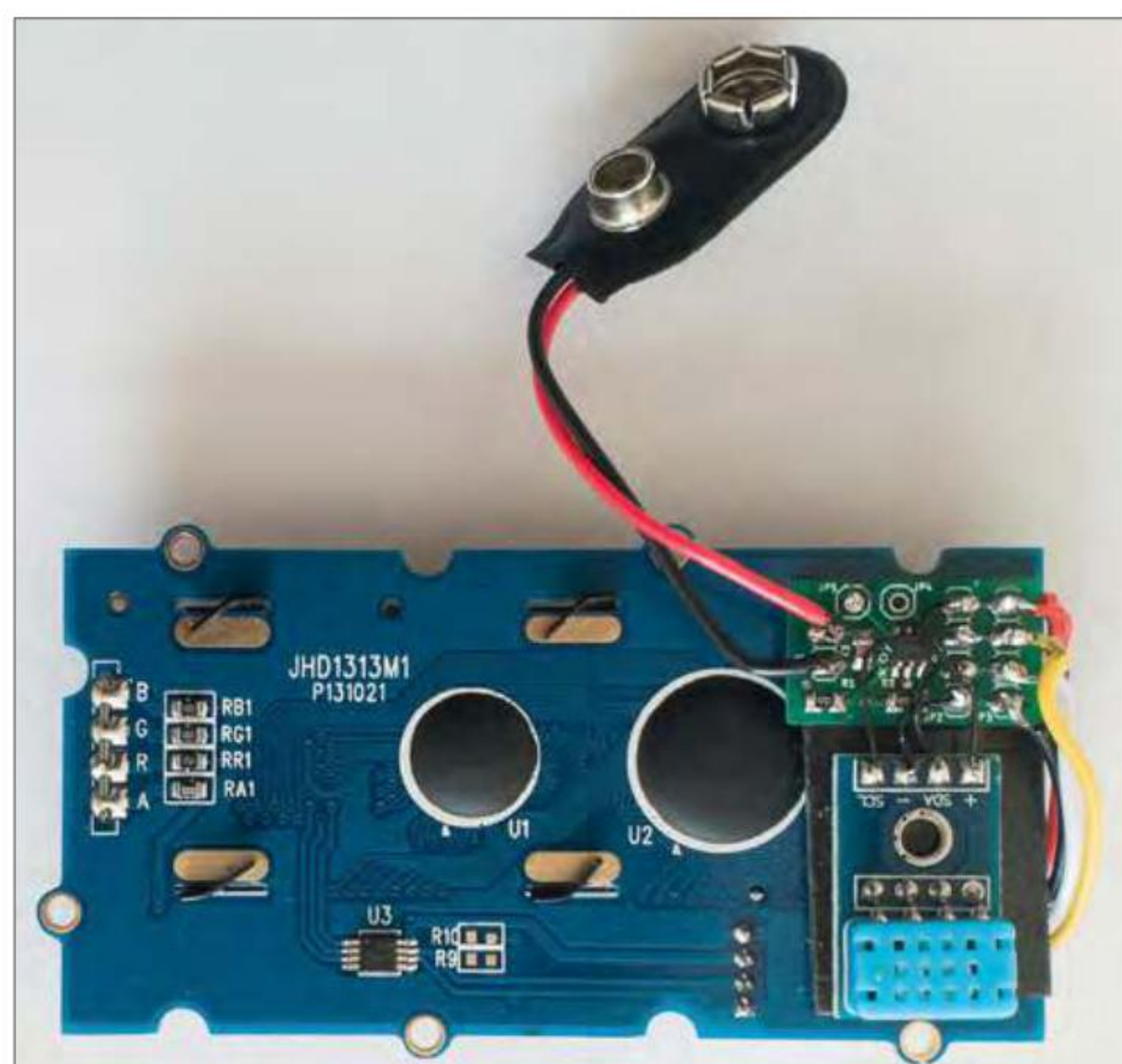


Fig.8. Sensor, PCB and LCD final assembly.

AUDIO OUT



By Jake Rothman

PE Mini Monitor Crossover – Part 1

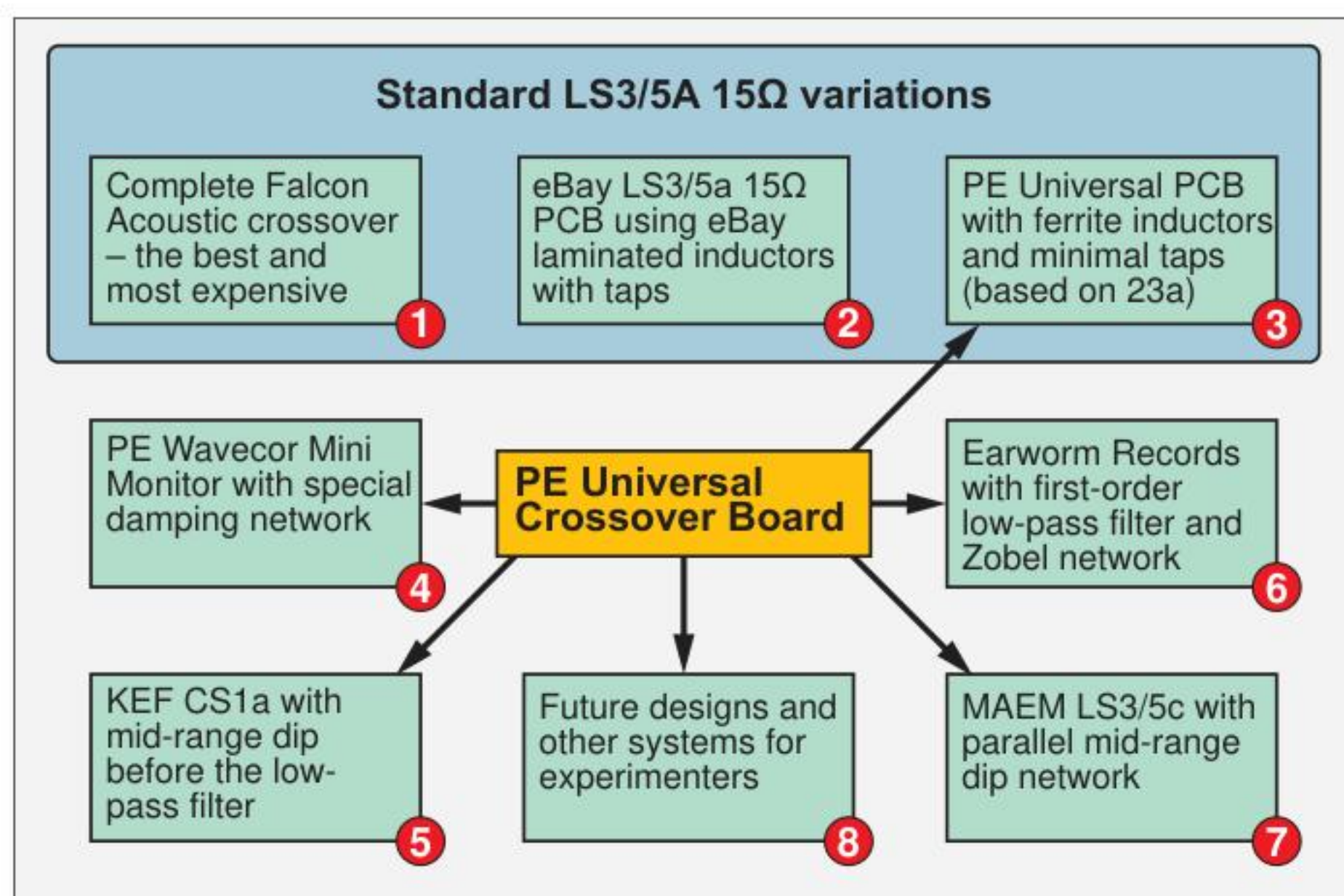


Fig.1. Diagram showing the possible crossover designs in this series of articles. Note the multitude of two-way crossovers that can be built on the universal board.

Welcome to the next stage in our LS3/5a journey. We were going to cover the enclosure this month, followed by a nice choice of crossovers, but we've actually made more progress with the latter, so we'll cover the all-important crossover first and then the enclosure hardware. It's surprising how complicated a simple wooden box can be, but we want it to be flexible enough for a whole range of future designs, as well as the LS3/5a, so it's worth taking the time to get it right.

Introduction

This project is not just about 'building an LS3/5a', it's also about providing you with the foundations to use a variety of drivers and associated crossover designs based on a single cabinet and (mostly) a single crossover PCB. Fig.1 shows the crossover options. Versions 1 and 2 are the exception to the above universality rule. Version

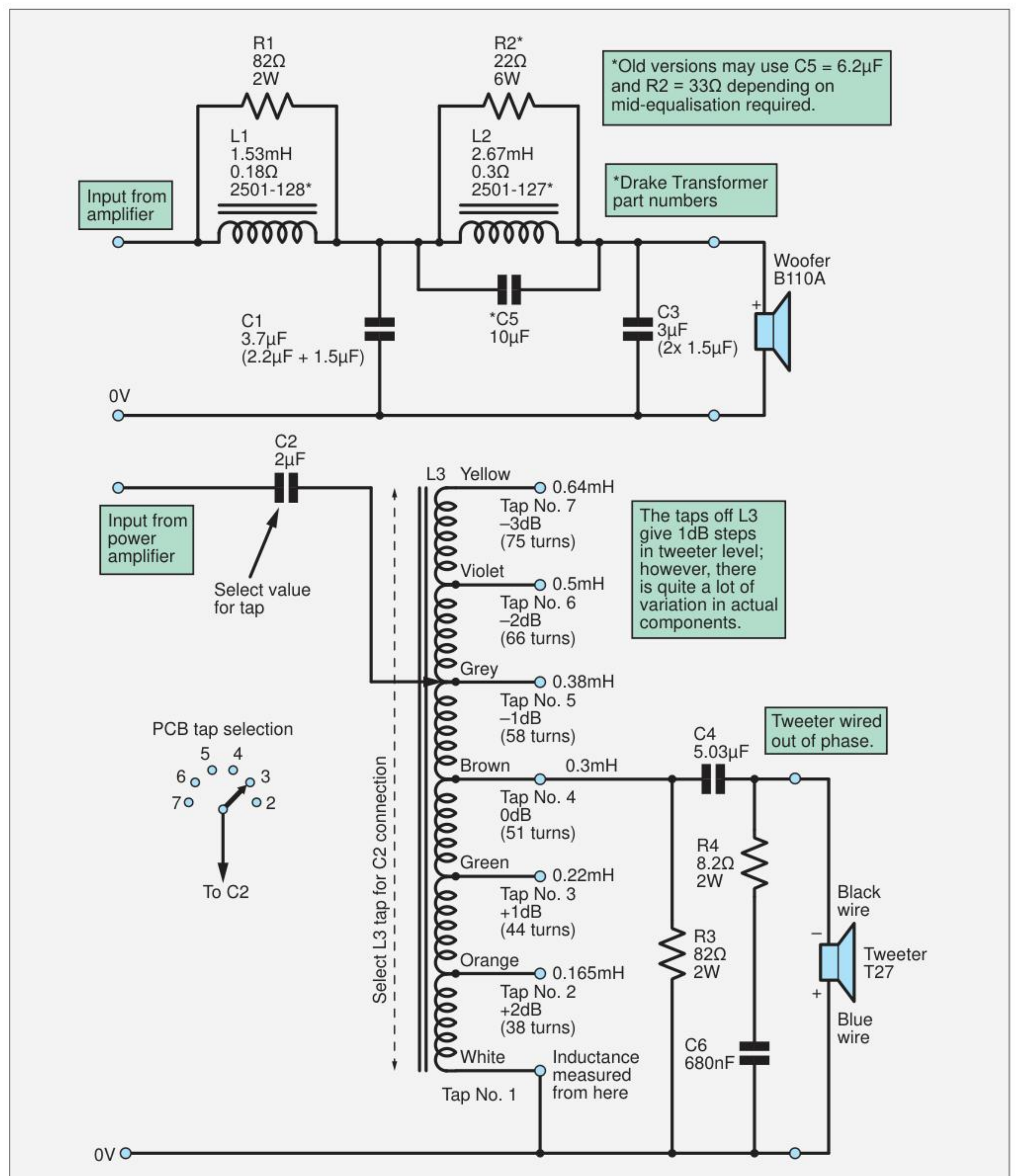
Fig.2. Original BBC 15Ω LS3/5a circuit. The tapped inductor is designed to vary tweeter level by 1dB steps. (Note that in the Fig.33 of the October 2019 article I set the tap steps at 0.5dB; however, Malcolm Jones of Falcon Acoustics advised that the taps should actually be 1dB steps. That said, in my tests of numerous inductors in various LS3/5a varieties there is a great deal of deviation – inductors do have huge tolerance values.)

1 is simply the official LS3/5a crossover design from Falcon Acoustics. It is of course excellent, but it's expensive and really only designed for just the LS3/5a speaker. Version 2 is the budget eBay Chinese version of this. It uses an eBay PCB and eBay laminated inductors. The results are good, but inevitably the cheapness of components results in a 'budget' version. Next are Versions

3-7 (plus future potential varieties). These use our own *Universal Crossover Board*, a very flexible design with bags of potential for tweaking and modding. It avoids the expensive 'official BBC' inductors by using high-quality ferrite-core devices.

This month, we will build Versions 2 and 3 (Version 1 comes to you direct from Falcon Acoustics). Next month, we will cover the Wavecor design.

So, there are several basic designs of crossover for the *PE Mini-Monitor* projects. The first one we are going to discuss is the classic BBC standard 15Ω LS3/5a circuit shown in Fig.2 (this is an edited version of Fig.33, *Audio Out*, *PE* October 2019).



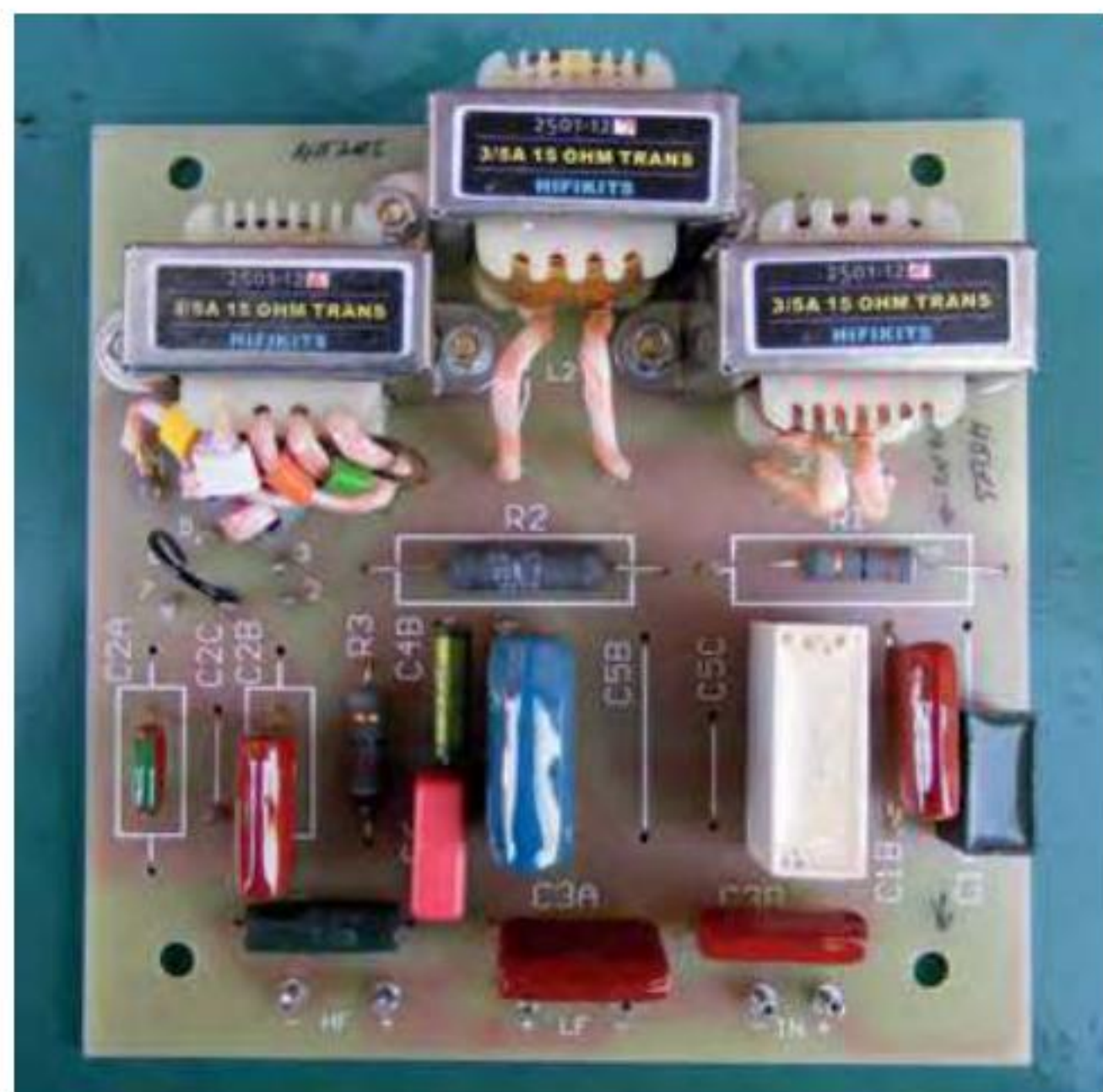


Fig.3. LS3/5a 15Ω crossover board from eBay, but for how long? The component legends correspond to the BBC circuit.

Version 2: classic BBC crossover

This version uses an LS3/5a 15Ω board available on eBay, as shown in Fig.3. It uses the B110A and T27 drivers from Falcon Acoustics or the large second-hand component market (as always, see eBay). A matching laminated inductor kit is available on eBay. These Chinese coils don't quite meet the BBC component tolerance specification of $\pm 5\%$ (more like -7%) but they certainly do the job. They are shown in Fig.4.

Components – Crossover Version 2

All component values are $\pm 5\%$.

PCB

From eBay – eg, eBay part 191278013041, or search for 'LS3/5a crossover 15ohm'.

Resistors

(WW = wire-wound)

R1, R3 82Ω 2W, WW or metal-oxide

R2 22Ω 6W WW (*see next month's tweaks*)

R4 8.2Ω 6W WW

Capacitors

The original BBC design used metalised-polycarbonate (MKT) types, such as STC/ITT PMC, Wima MKB3, and Advance Filmcap, all of which are no longer



Fig.4. Laminated LS3/5a inductors are available on eBay. The manufacturer does not state if they use Radiometal or silicon-iron, but they sound fine.

made (although Falcon had some polycarbonate capacitors specially produced). I use polyester or MKT, which are the nearest in sound quality, but they're not as stable. (Rogers always used the clear-epoxy-cased Rifa PHE type which were MKT.) All devices are radial mounting, minimum voltage 63V, although using higher voltage 100V and 160V gives slightly lower distortion.

C1 3.7μF use 2.2μF (C1a) + 1.5μF (C1b)
C2 Value depends on inductor tap selected for tweeter level to match woofer. Normally 1.72μF with tap No. 6 (violet) for post-1984 T27 tweeters. Composed of C2a 1.5μF and C2b 220nF. (For other taps, C2 value differs – see table bottom of page.)
C3 3μF (2 × 1.5μF)

C4 5.03μF (4.7μF + 330nF)

C5 10μF (may need to be reduced for some B110As – see next month)

C6 680nF

Inductors

Set of six (for a pair of crossovers) via eBay from Hifikits – eBay part 323926563397, or search for 'Inductor for LS3/5a 15ohm Crossover/Frequency divider One Pair(6 Pieces)' – yes, 'ivider' is correct at the time of writing! Price is around £65 delivered.

L1 1.53mH, direct current resistance (DCR) 0.18Ω or less. Laminated Radiometal, or similar 45 to 49% nickel-steel alloy. Core size 40mm long × 35mm × 10mm wide stack. Fixing centres at 46mm, 3.8mm holes.

L2 2.67mH, DCR 0.3Ω or less, same core as L1.

L3 0.64mH tapped at 0.5, 0.38, 0.3, 0.22, and 0.165mH, DCR 0.13Ω or less, same core as L1.

Hardware

- To secure the laminated inductors, use six M3.5 Pozi cheese-head screws with nuts and lock washers
- Seven 0.15-inch and seven 0.1-inch Vero pins are needed for the tapping points, plus six 0.2-inch pins for the terminals
- For PCB mounting, M5 × 40mm countersunk bolts to attach the PCB to the speaker front baffle; 12 nuts and lock washers are also needed.

Construction – Version 2

Assembly is very straightforward, in fact much easier than most PCBs, since the components are large – more like a 1960s radio. First, put in the terminal pins, then mount and solder the inductors and finally the resistors and capacitors. The thick tapping wires on inductor L3 are composed of two pieces of wire and these won't go through the board holes so they are soldered to pins on top of the board.

We haven't supplied an overlay diagram as the board is very clearly marked and you can see large photos of it on page 51 of *Audio Out*, PE October 2019.

Tweeter attenuation

In most speaker systems the woofer is less sensitive than the tweeter, with less acoustic output emitted for a given electrical input power. Sensitivity is usually specified in dB per watt at 1m. Tweeters tend to have a wide tolerance on this parameter, being more influenced by magnet strength and diaphragm mass variations than woofers. This has to be accounted for in the crossover design to ensure consistency of the finished speakers' frequency response. Normally, this is done by changing a resistor feeding the high-pass filter. The BBC LS3/5a and other BBC designed speakers used a more complex solution, an autotransformer with taps. This transformer also doubled as the high-pass filter inductor. In a factory, the tweeter level is set by measuring the frequency response. For home constructors, it normally has to be set by ear. For most people with reasonably musical hearing it is usually fairly obvious if the tweeter is too loud or too quiet (relative to the woofer). Of course, a well-balanced source of music has to be used.

Tweeter attenuation on the BBC crossover

In the original BBC LS3/5a crossover the tweeter attenuation is set using taps on L3; you can see this in Fig.2. On the original BBC and eBay PCBs there is a small semicircle of six Vero pins numbered 2 to 7, one of which has to be connected via a link to the pin at the centre of the semicircle to set the attenuation level. On the original BBC and Falcon inductor these taps are also colour coded – unfortunately not in resistor colour code – and these colours are listed below.

As the attenuation is increased from tap 2 to 7, the input impedance goes up because L3 is a transformer, so to maintain the same filter curve, the input capacitor C2 has to be reduced correspondingly. It's important to understand that the value of C2 depends on the actual tapping used on L3. Therefore, it's best to have on hand all the possible capacitors required for the attenuation tapplings to match the tweeter to the crossover. The capacitor values required for each tap are:

Tap 2 orange (maximum boost) 5.5μF (3.3μF and 2.2μF)

Tap 3 green 3.98μF (3.3μF and 680nF)

Tap 4 brown (theoretical unity gain) 3μF (1.5μF × 2)

Tap 5 grey 2.3μF (2.2μF and 100nF)

Tap 6 violet 1.72μF (1.5μF and 220nF)

Tap 7 yellow (minimum volume) 1.36μF (680nF × 2)

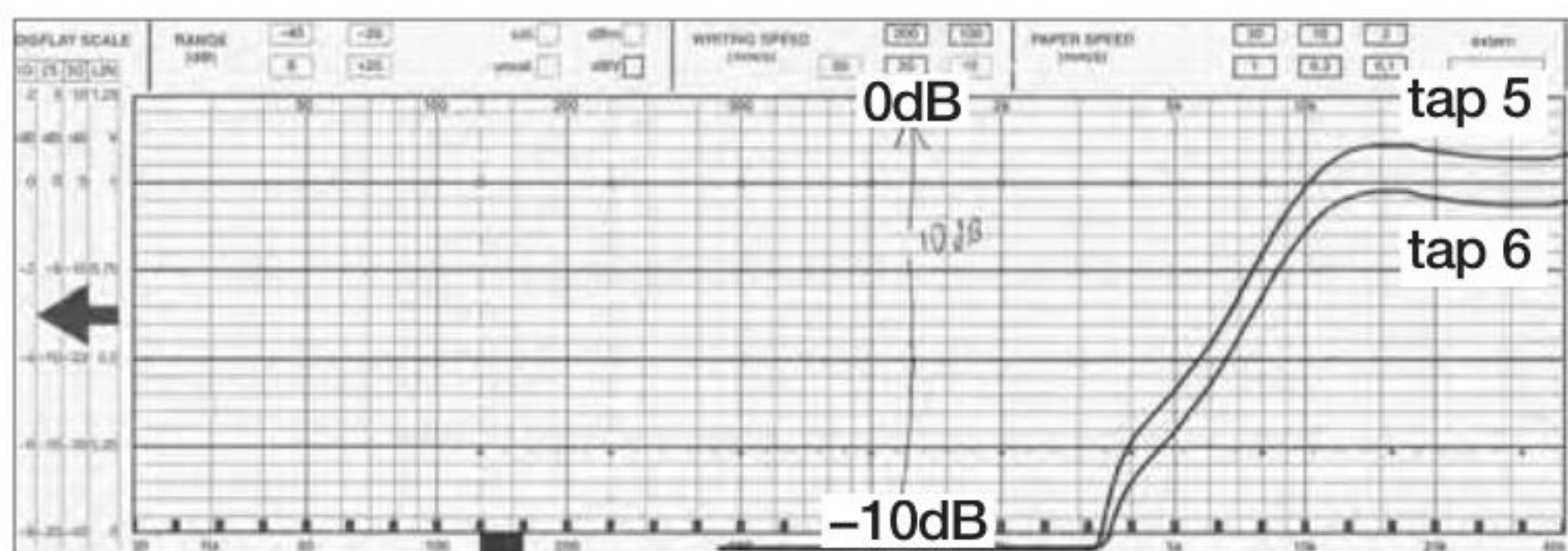


Fig.5. Tweeter crossover curve using Spondor tapped inductor. Note both curves are attenuated below the 0dB axis at top of the graph.

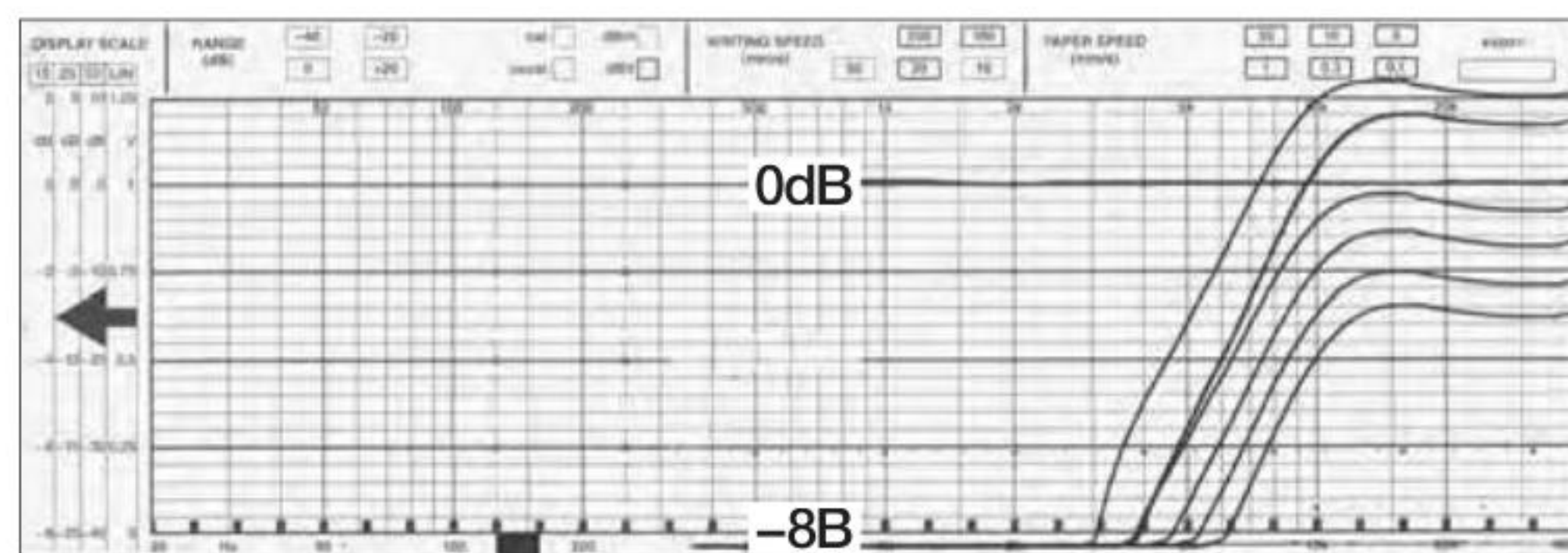


Fig.6. Tweeter tapping levels using an eBay inductor. Top to bottom, curves for taps 2 to 7.

In reality, it's very unlikely that the tweeter will have more output at the crossover point than the woofer, so the boost tapplings on numbers 2 and 3 are almost never needed. This just leaves the option of attenuation, and normally only a couple of dB is needed. There is already almost 10dB of mid-range attenuation/low-frequency boost equalisation from the low-pass section of the crossover feeding the woofer, which is also taken into account in matching the relative levels. In practice, taps 5 or 6 are the most frequently used in most LS3/5a crossovers. It's best to start with tapping 6, which means C2 is 1.5 μ F + 220nF in parallel.

We will see how we can simplify things on the *Universal Crossover Board*, where a much simpler (and cheaper) inductor is used.

Measuring inductors

Inductors are the least perfect electrical component and their inductance changes appreciably with frequency. In crossover design and construction its best to treat

quoted values with a degree of suspicion and actually measure what you have. The BBC specification states the inductance should be measured at 3kHz, approximately the crossover frequency. If you have an old inductance bridge the frequency can be set to 3kHz or whatever is desired. I tried using a Tema 72-6634 digital LCR meter which was fixed at 200Hz, which was too low. The Peak LCR45 impedance analyser, which has three switchable frequencies, gave good results at 1kHz but under read at 15kHz and 200kHz due to the coil's parasitic winding capacitance.

I measured an old two-tap Spondor inductor (just having taps 5 and 6) and an eBay one and generated the curves shown in Fig.5. The Volt-manufactured Falcon ones are apparently spot-on, but I am awaiting a sample to confirm this for myself. The Volt site has an interesting article on this inductor: <https://voltloudspeakers.co.uk/not-your-average-inductor>

I also measured the eBay inductors and their inductance values are shown below measured at 1kHz. I only had two of each, which is a small representative sample:

- L1** 1.62 and 1.58mH, 0.15 Ω rather than 1.53mH
- L2** 2.53 and 2.68mH, 0.28 Ω rather than 2.67mH
- L3** Tapped coil 0.586 and 0.597mH, 0.04 Ω , rather than 0.64mH.

The resulting tweeter crossover curves are shown in Fig.6. Notice that the two step-up taps (at the top) are excessive, but it doesn't matter, no one uses them anyway. The connections to the inductor are shown in Fig.7. I put coloured cable number sleeves on mine to avoid confusion before soldering to the board.

Version 3: 15 Ω LS3/5a with 23a high-pass filter using Universal Crossover Board

Now we move on to our super-flexible *Universal Crossover Board*. The eBay LS3/5a board is specifically designed for the BBC LS3/5a crossover, but to implement the different versions we want to offer we have designed a new PCB that is much more flexible in configuration and the component choices it allows – particularly with respect to the inductors.

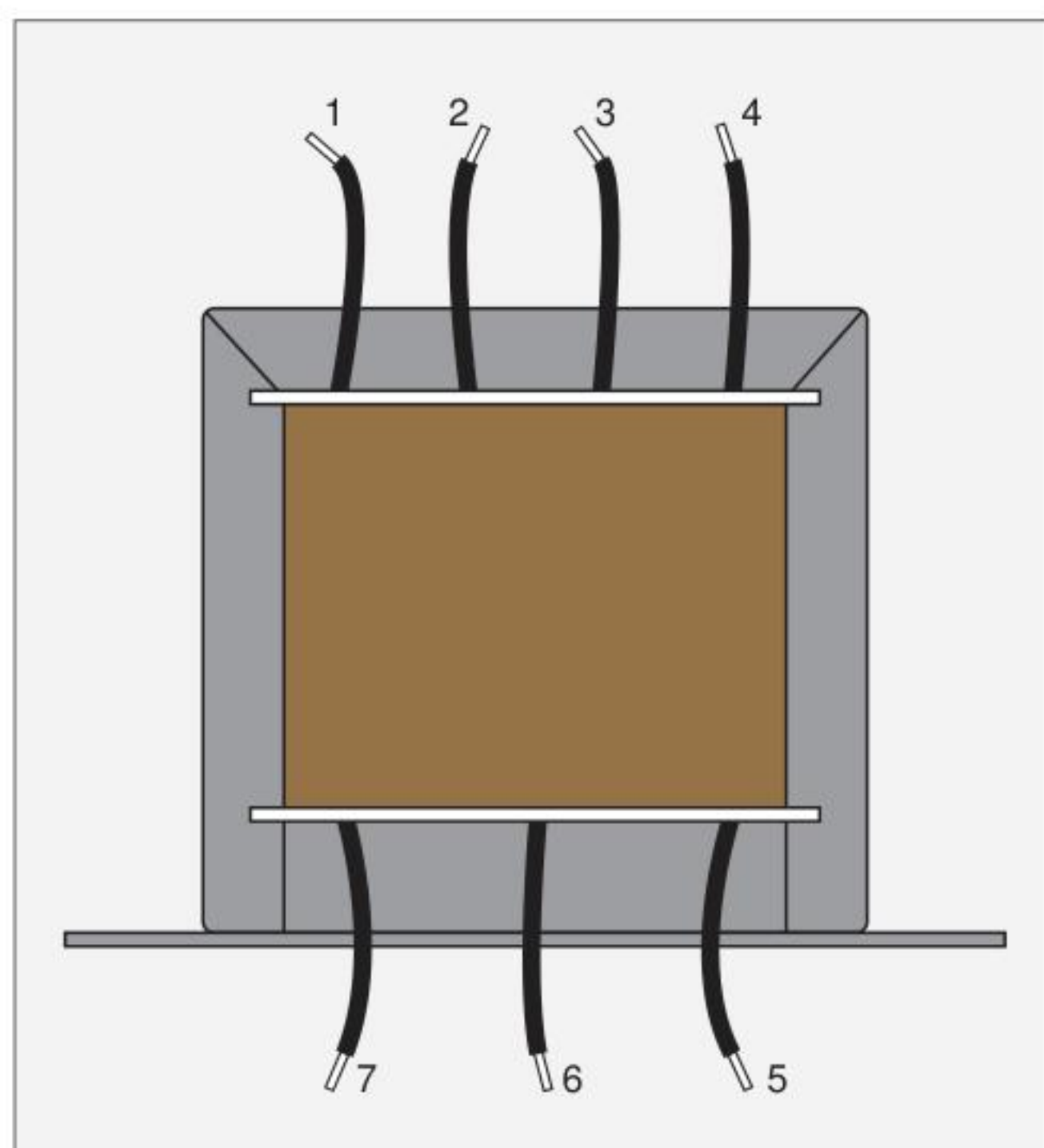


Fig.7. Connections to eBay tapped inductor

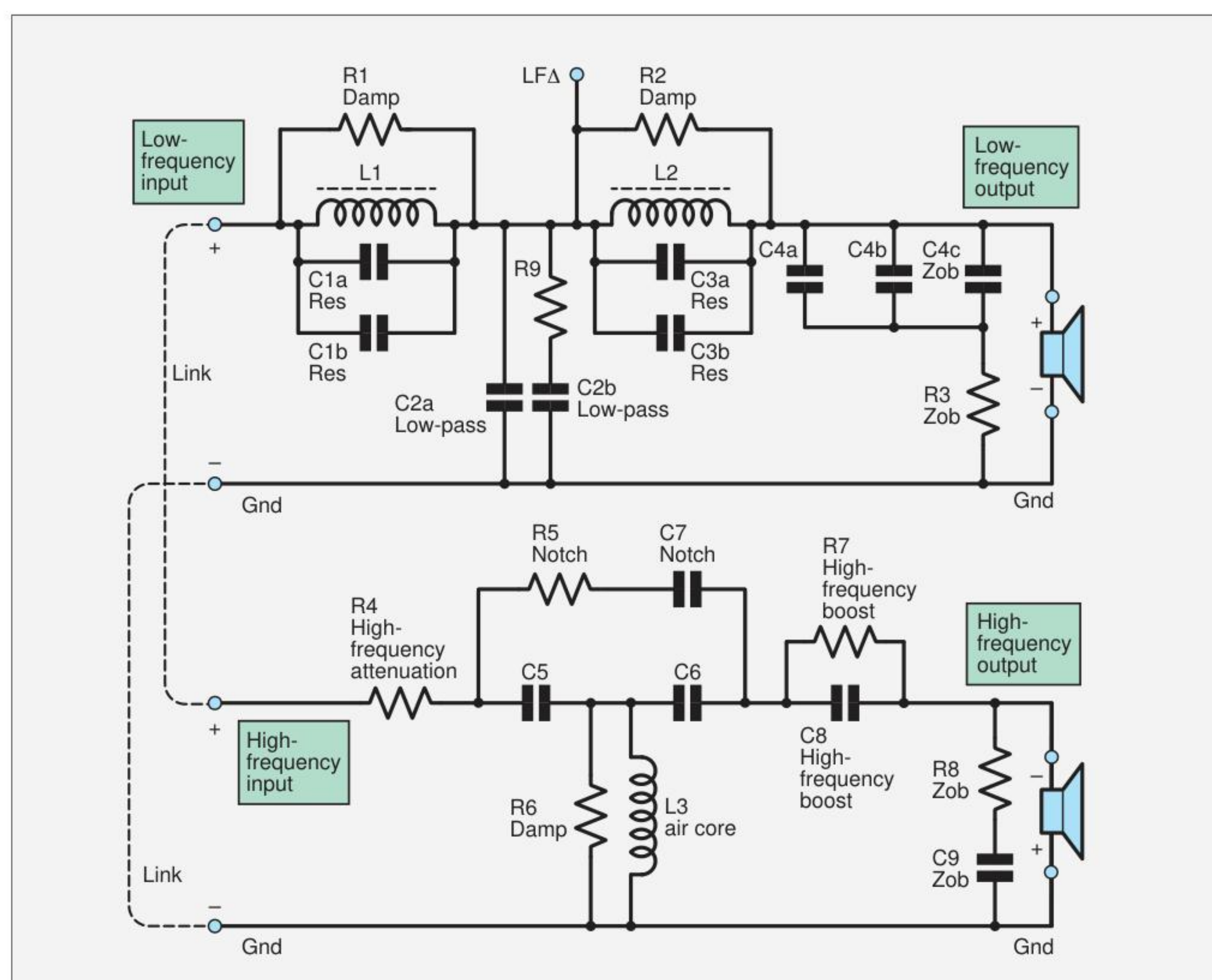


Fig.9. The full circuit possibilities of the universal board. Not all component positions are used for every design.

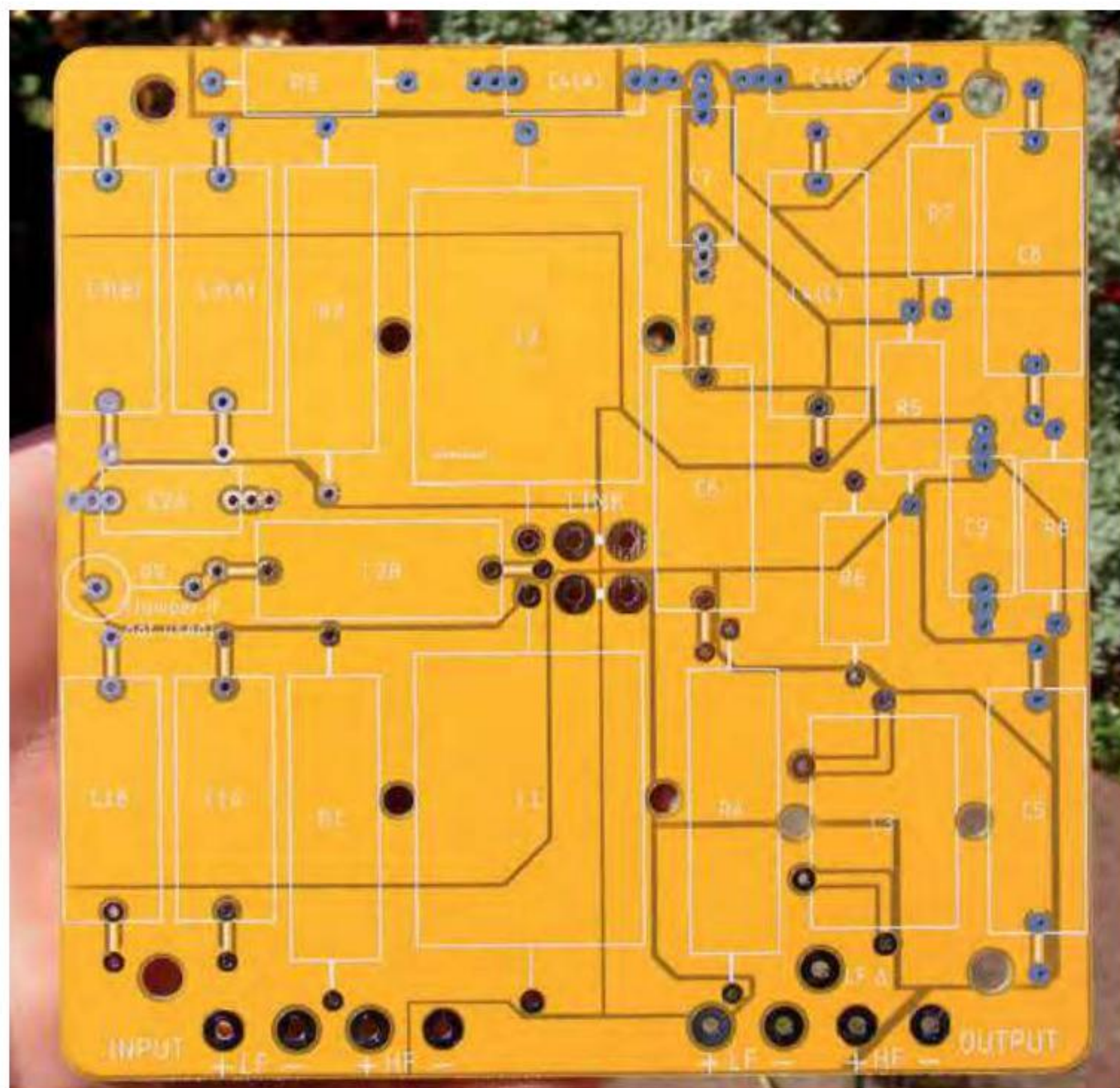


Fig.8. The PE Mini-Monitor Universal Crossover Board.

Plus, we all know how insecure eBay supplies can be – here today, gone tomorrow – so for all the above reasons I commissioned Mike Grindle to create a *Universal Crossover Board* PCB to my design, shown above in Fig.8. It will be available from the *PE PCB Service*.

This PCB uses ferrite inductors, which are available from many suppliers such as Volt, Falcon, Wilmslow Audio and Cyber Market. It is a versatile board that can accommodate many standard two-way crossover circuits, summarised in Fig.1. (It also replaces the Falcon PCB7 board used for the 23a version, which is no longer listed.)



Fig.11. The board has the option for bi-wiring. If not used then the woofer and tweeter sections are linked as shown above.

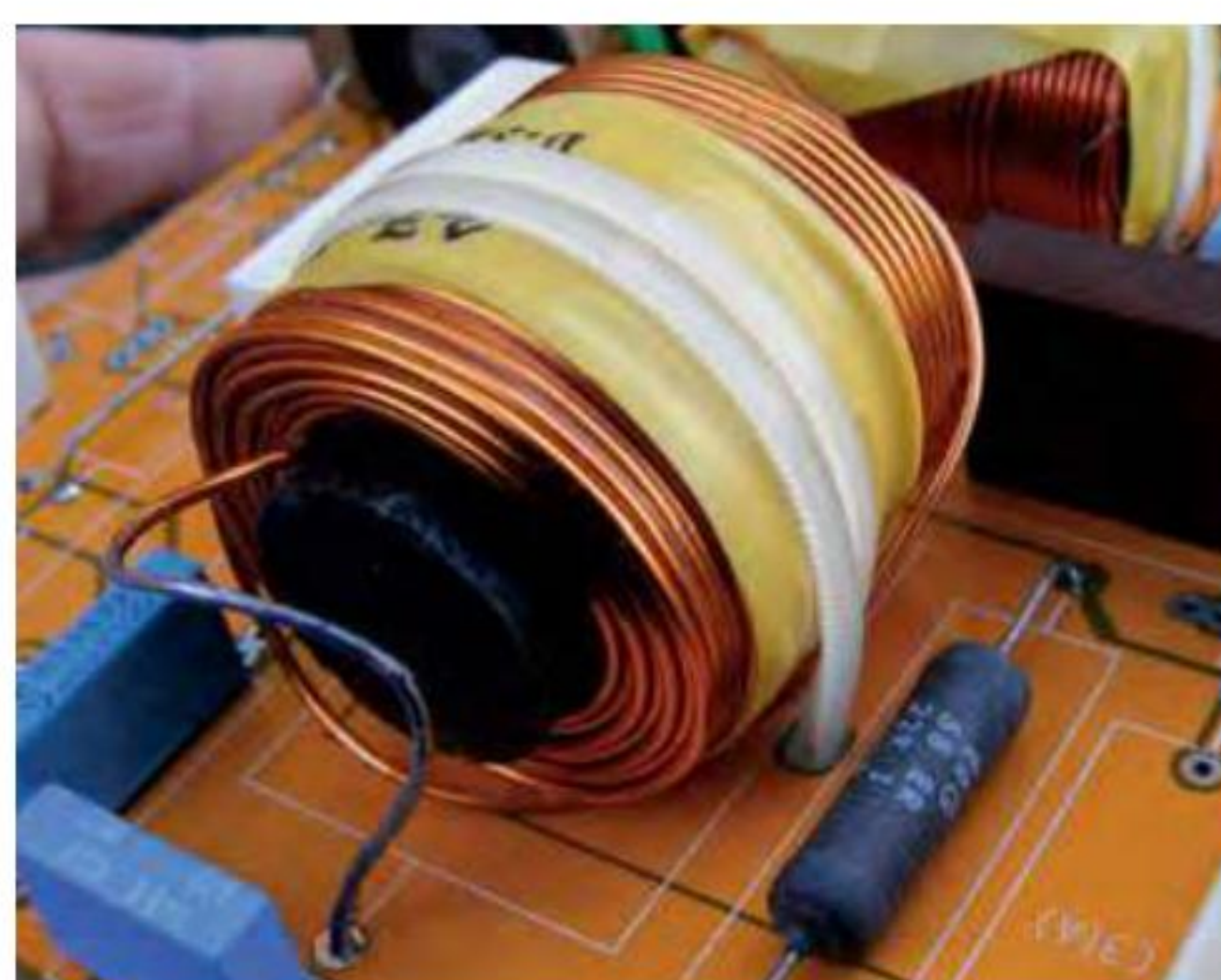


Fig.12. Heavy inductors must be supported with cable ties or the wires may break.

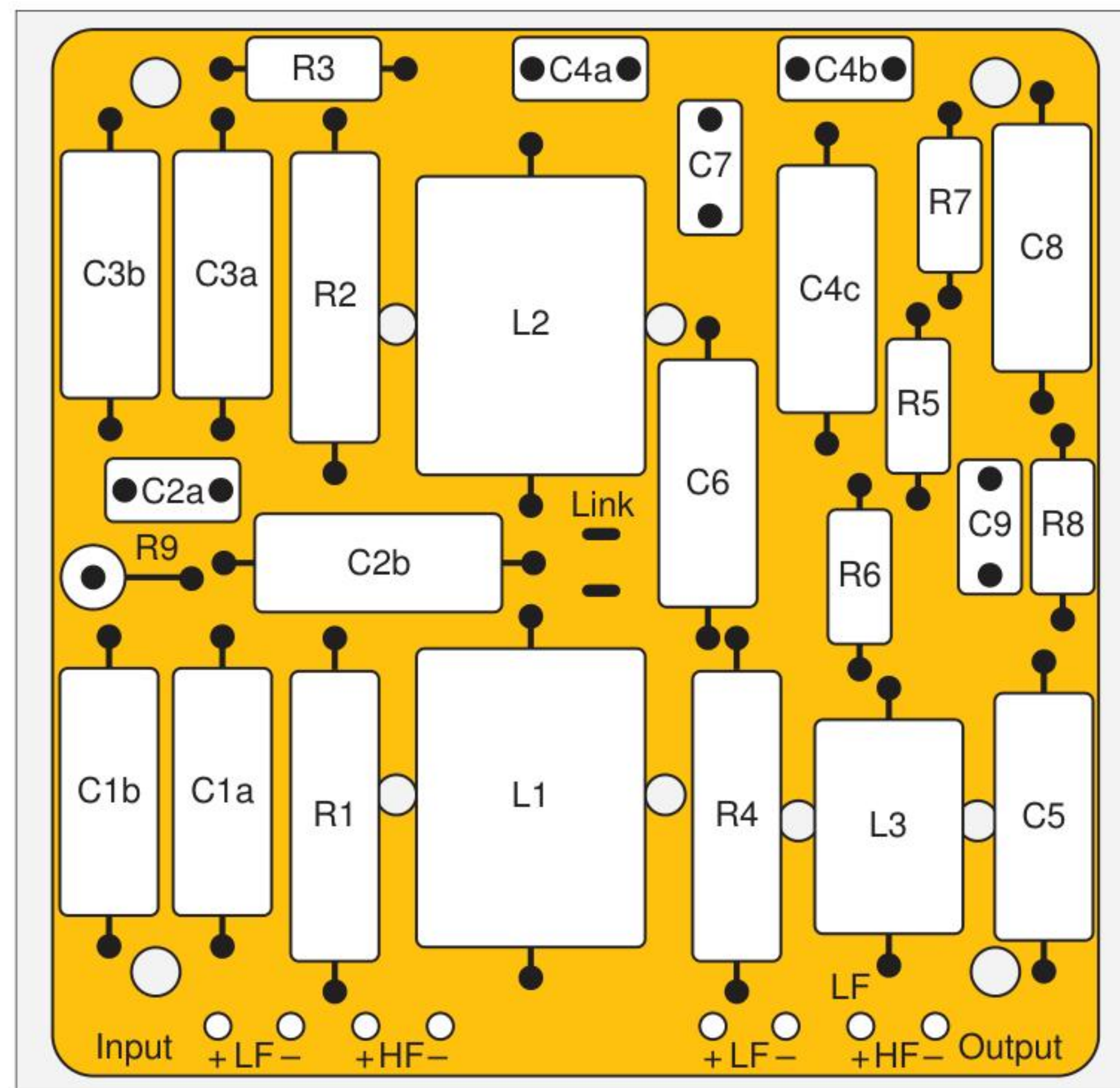


Fig.10. Overlay for PE Mini-Monitor Universal Crossover Board.

The *Universal Crossover Board* can be set up for all the standard two-way configurations. Options for up to third-order filters, parallel and series-tuned circuit mid-dip networks, damping resistors and Zobel networks are available. It uses high-power ferrite inductors in the LF section and a small ferrite (or optionally air-core) in the HF section. It is also ideal for developing one's own designs. I'll give details for building a few different versions, all using the same highly flexible board. The dimensions and fixing holes are to BBC specification so it has to be fitted on the back of a removable front baffle or rear panel.

For the *Universal Crossover Board*, if large air-core or laminated inductors are used then these will have to be mounted with flying leads off the board. The general circuit is shown in Fig.9, and the general overlay in Fig.10. Provision for bi-wiring is provided so that separate wires can be used to feed the bass and treble sections, labelled on the board 'LF' and 'HF' respectively). If bi-wiring is not wanted, the two sections can be joined together by the links in the middle of the board (Fig.11). On the PCB, multiple outlines allow for different capacitor sizes, including axial types. Provision is also made for capacitor paralleling to

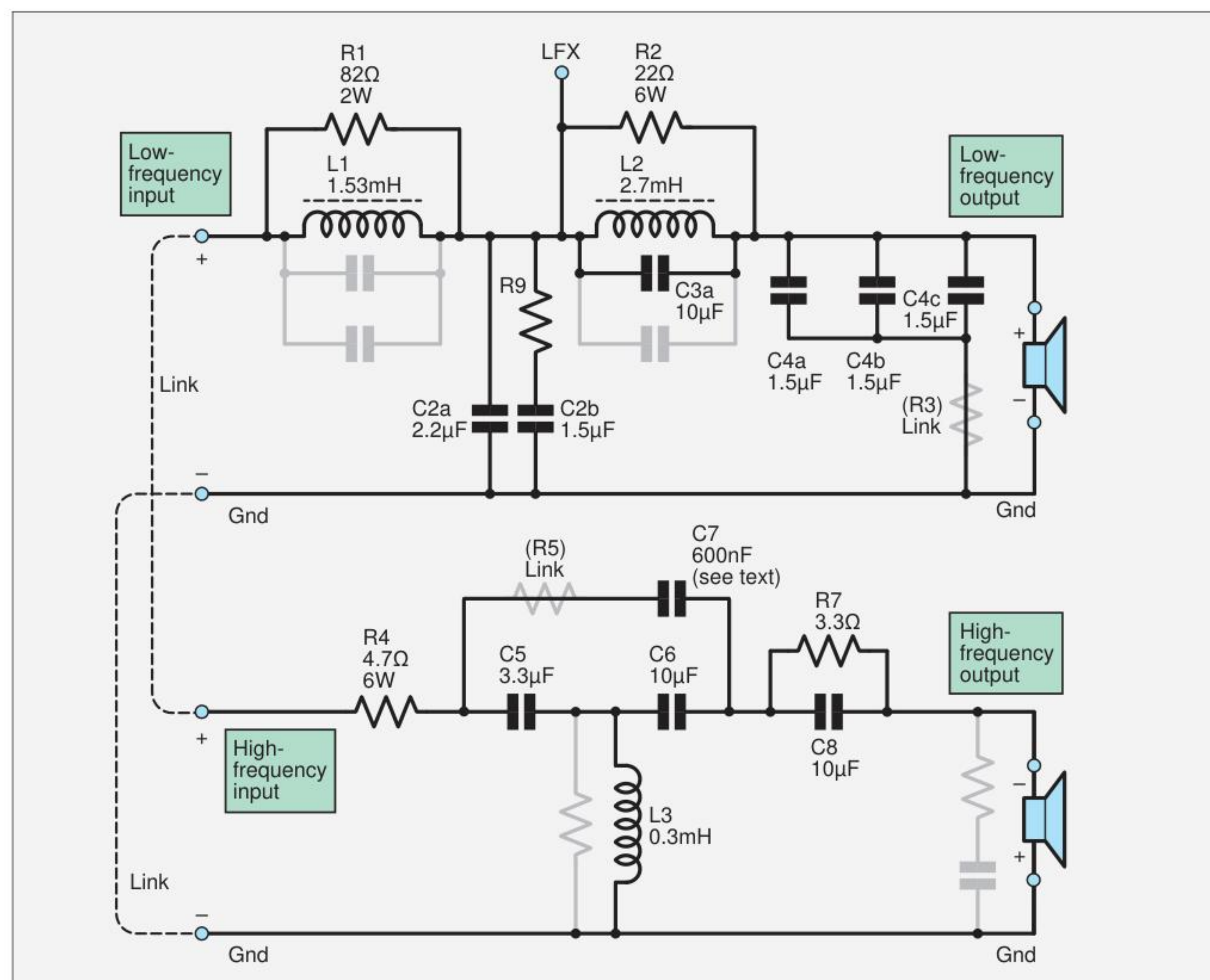


Fig.13. Components (highlighted) used for the LS3/5a 23a crossover built on the *Universal Crossover Board*. Note: part numbers correspond to the circuit in Fig.9.

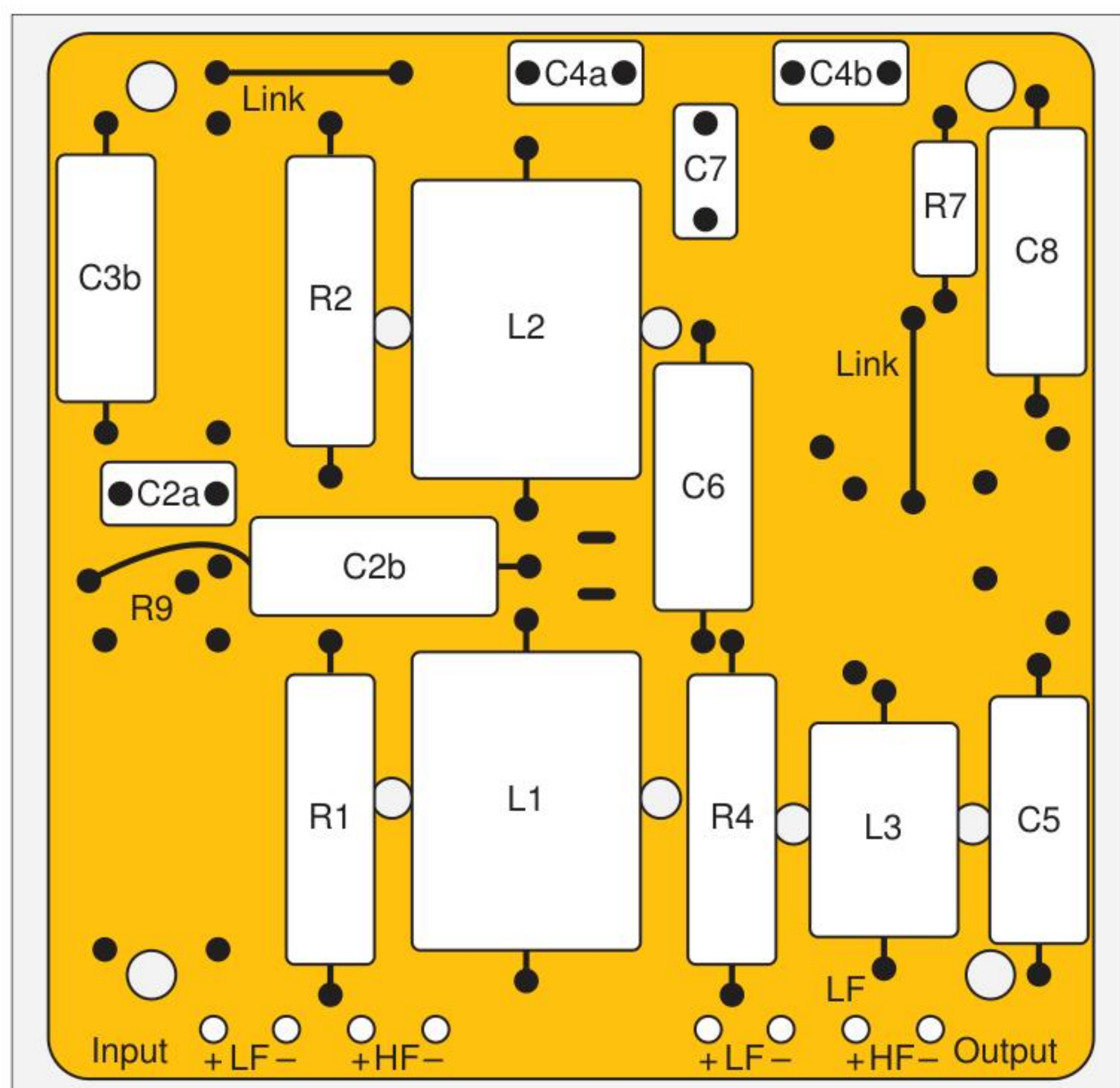


Fig.14. Overlay of components for Version 3: 15Ω LS3/5a with 23a high-pass filter using the *Universal Crossover Board*.

obtain large/non-standard values. For example, C4 can be up to three capacitors C4a, C4b and C4c.

Lower-cost coils

The low frequency ferrite-cored inductors (L1 and L2) should be high-power types, 50mm × 12.5mm and wound with 1mm wire to get the DC resistance as low as in laminated core inductors. A smaller 25mm core can be used for the tweeter inductor (L3). Because of their weight, these inductors must be supported with pairs of cable ties threaded through the 3mm holes in the board (Fig.12). The 'exact' LS3/5a inductor values can be bought from Falcon, but given the tolerances involved, standard E12 series values of 1.5mH and 2.7mH are fine. To remove the requirement for the expensive tapped inductor (L3), the high-pass section uses the Falcon 23a circuit, which has just a simple 0.3mH inductor and a resistor (R4) for tweeter attenuation.

Components – Crossover Version 3

For building the LS3/5a 15Ω 23a circuit on the *Universal Crossover Board*, not all component positions are used (remember this is a board offering many options), thus the component numbering is changed compared to the BBC standard in Version 2 above.

Version 3 relates to the universal circuit diagram in Fig.9 and the components used are highlighted in Fig.13. Note that some components positions (R3, R5 and R9) have to be linked with 20swg tinned copper wire. The Version 3 components (but NOT their part numbers) are the same as in the low-pass filter for Version 2, apart from the use of ferrite inductors. The overlay is shown in Fig.14 and

a completed 15Ω LS3/5a 23a board is shown in Fig.15. For C2b, a 2.2μF axial capacitor can be stretched out far enough to link out R9, avoiding the link shown in Fig.16. The tweeter level is now set by R4, and can range in value from 2.7Ω (loudest) to 6.8Ω (quietest).

All component values are ±5%. Note that components numbers refer to Fig.9, not Fig.2.

PCB

Available from the PE PCB Service. Component kits will also be available when the cabinet kits are released.

Resistors

R1 82Ω 2W WW or metal oxide
R2 22Ω 6W WW
R3 *linked*
R4 4.7Ω 3W WW (select for tweeter level)
R5 *linked*
R6 *not used*
R7 3.3Ω 3W WW
R8 *not used*
R9 *linked*

Capacitors

All 63V minimum, 5% polyester
C1a, C1b *not used*
C2a, C4a, C4b 1.5μF
C2b 2.2μF
C3a, C6, C8 10μF
C3b *not used*
C5 3.3μF
C7 600nF (use a low 680nF)

Inductors

L1 1.5mH 50mm × 12.5mm ferrite core, DCR 0.3Ω (Volt part F022)
L2 2.7mH 50mm × 12.5mm ferrite core, DCR 0.43Ω (Volt part F079)

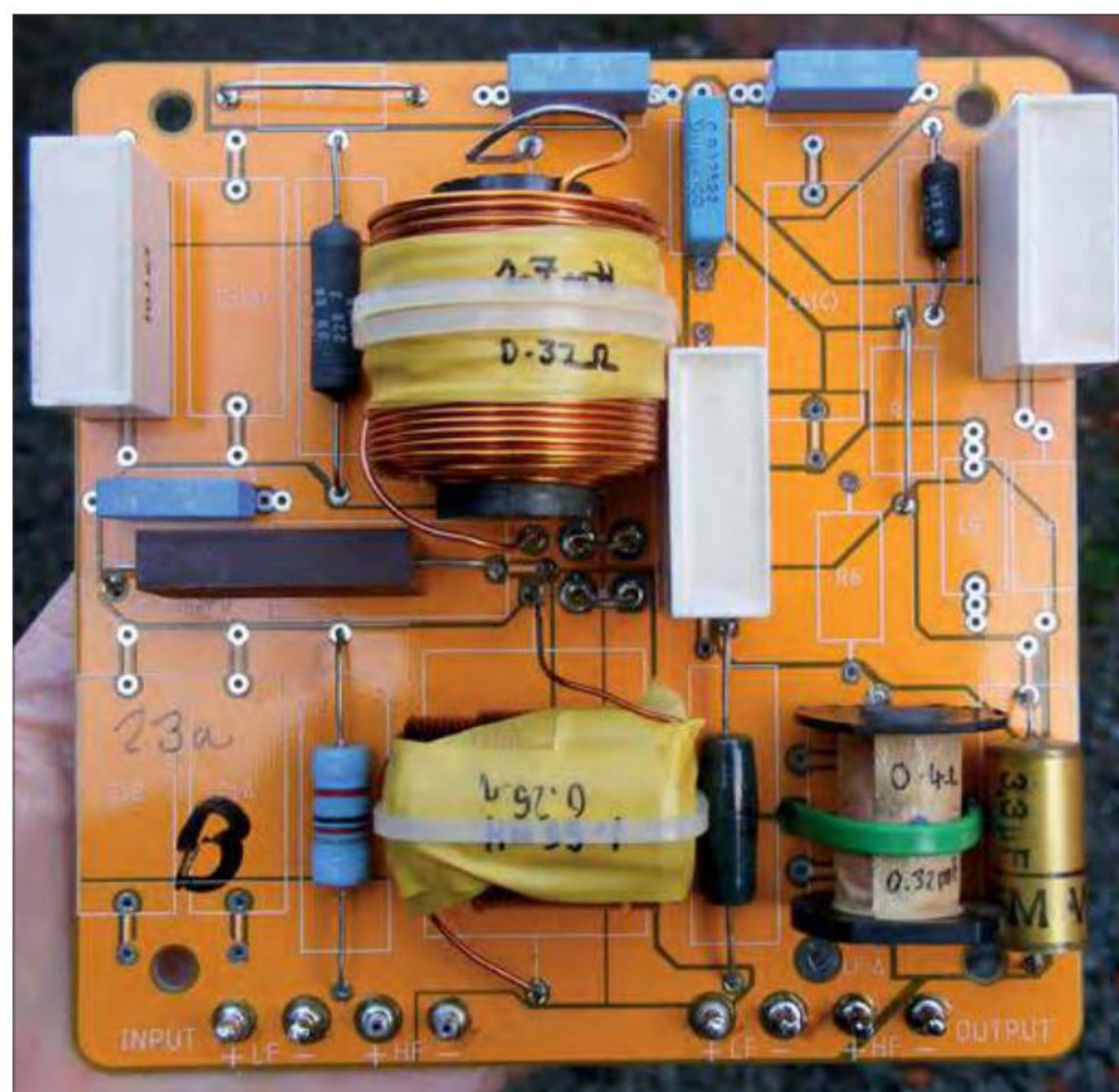


Fig.15. Completed Version 3: 15Ω LS3/5a with 23a high-pass filter using the *Universal Crossover Board*.

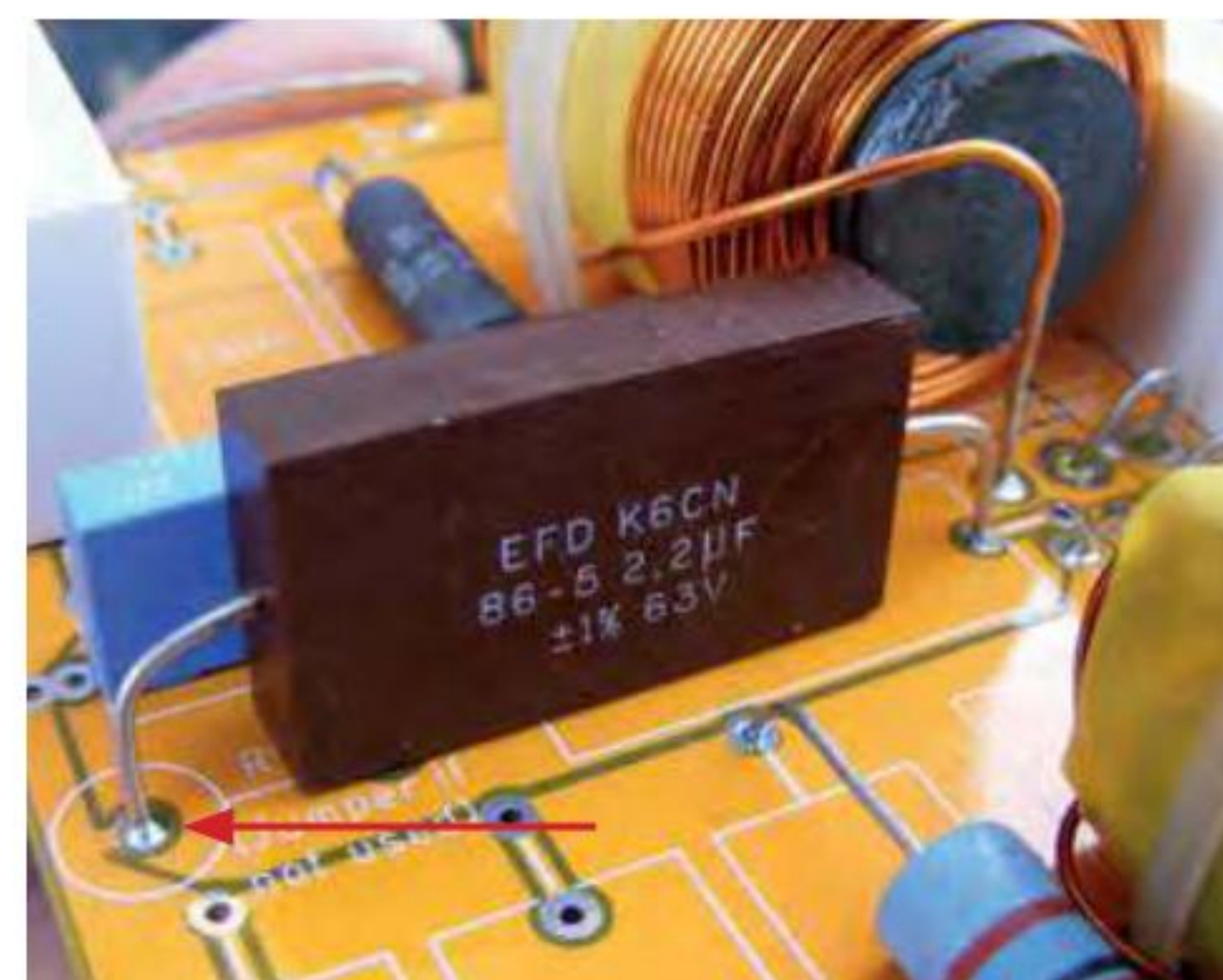


Fig.16. Avoiding using an R9 link by extending C2's wire to the R9 hole.

L3 0.3mH air-cored if possible for lowest distortion, but ferrite can be used, DCR 0.4Ω–0.5Ω (Volt part A0235)
www.voltloudspeakers.co.uk

Hardware

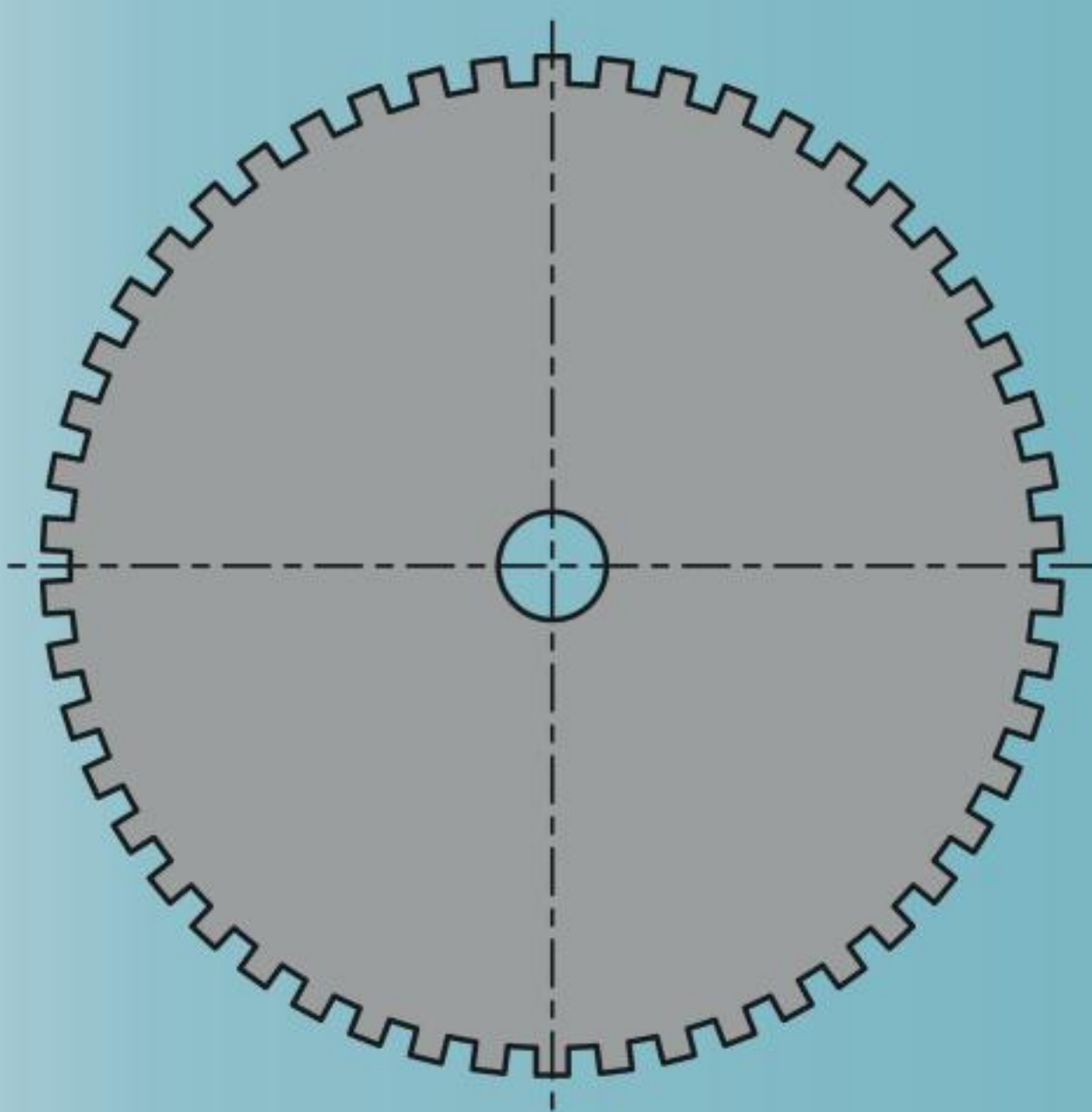
100mm × 3mm cable ties, 6 off
12 × 2mm terminal pins
100mm 20swg tinned copper wire for links

Construction – Version 3

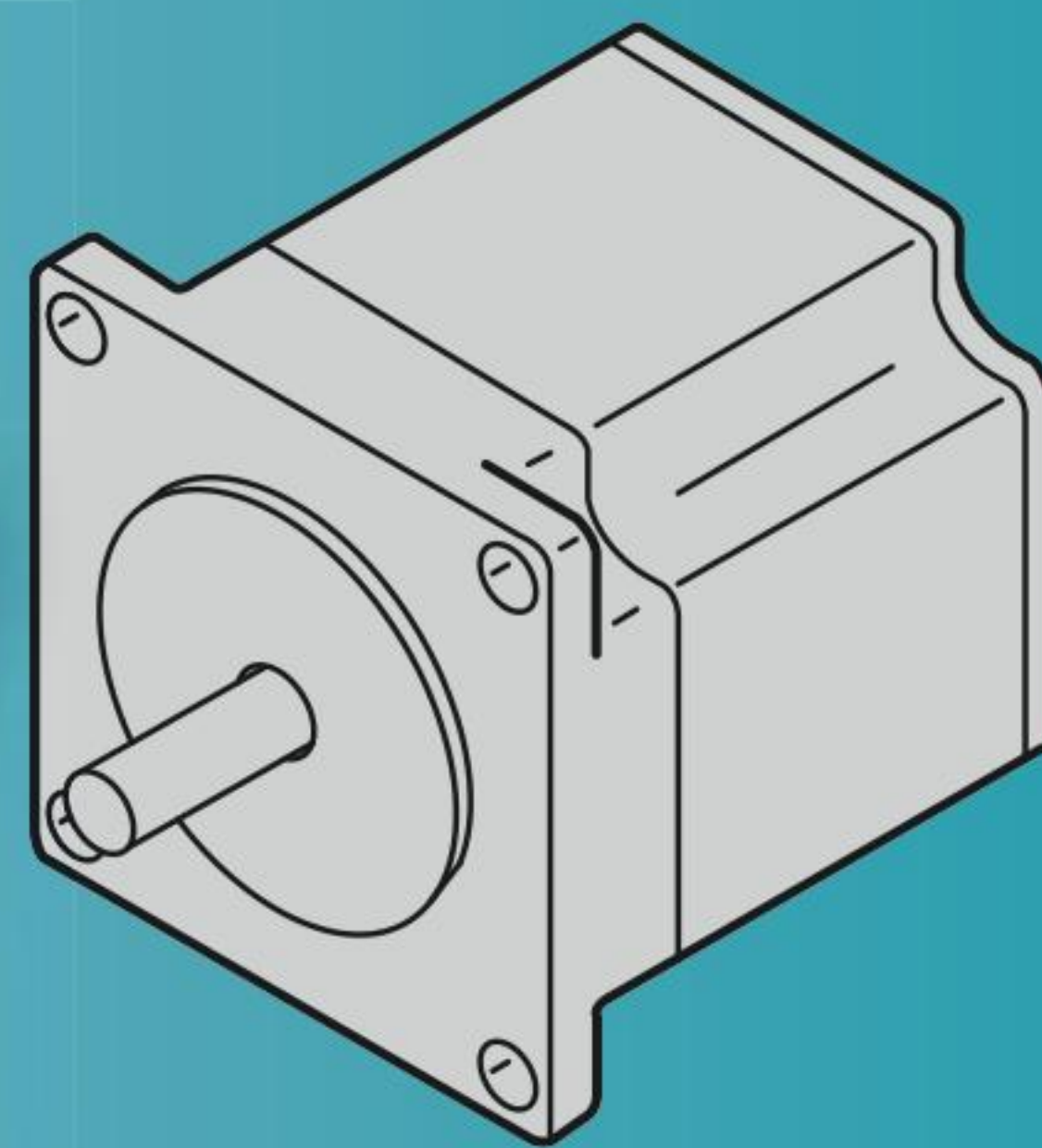
Assembly is straightforward; the components are large by modern standards. Ensure they are rigidly fixed down to prevent movement. Insert the terminal pins and links first. A melted capacitor case looks unprofessional and it can easily happen when trying to get to inaccessible pins. A higher power iron (at least 45W) is needed because of the large amount of PCB copper compared to standard PCBs

Next month

In the next issue we'll describe using the *Universal Crossover Board* with Wavecor drivers, some further tweaks for the LS3/5a and some measurement tips.



Using Stepper Motors



Part 4 | by Paul Cooper | technobotsonline.com

Bipolar stepper motor drivers

In this month's instalment of learning to use stepper motors we cover bipolar stepper motor drivers. For a given frame size, *bipolar* stepper motors generally offer higher torque and better efficiency compared to the *unipolar* motors described in the December 2019 issue of *PE*. However, these benefits come at a price because bipolar motor drivers are more complex in design compared to unipolar. Fortunately, that premium can be offset by the abundance of low-cost commercially available driver chips and modules.

The fundamental requirement for a bipolar motor driver is that it can reverse the direction of current flow in each of the two windings. This is achieved using a pair of 'H-bridge' circuits.

H-bridge

In principle, designing an H-bridge is relatively straightforward, that is until you actually try it and 'pop' no end of MOSFETs. In its simplest form, an H-bridge is just four solid-state switches arranged in an 'H' configuration, which will be familiar to many readers. Fig.27 shows the basic arrangement of the H-bridge – a circuit topology that is not only found in stepper motor drivers

but also widely used for reversing in DC motor controllers.

The references in Fig.27 to 'high-side' / 'low-side' switches is a terminology commonly used for H-bridge drivers. The high-side switches connect to the positive supply rail and the low-side switches connect to the negative or ground rail, more on this later. To get a DC motor to spin in one direction, close the left high-side and right low-side switches, as shown in Fig.28. To reverse the direction, close the right high-side and the left low-side switches (Fig. 29). The same methodology is used for reversing the current in a stepper-motor winding.

Let's replace the switches with MOSFET transistors, p-type for the high side and n-type for the low side (Fig.30). N-type MOSFETs are preferred as they tend to have a lower on resistance, which means lower power losses and cooler operating temperatures. Unfortunately, using n-types for the high side is not straightforward because the gate voltage has to exceed the bridge supply voltage. However, this can be achieved using a DC-DC charge-pump circuit to raise the voltage above the positive supply rail.

There is a potential problem here. If you were to close *both* the right-hand-side switches, or both the left-hand-side switches, then you would have a short circuit across the motor power supply. Of course, you could put measures in place that would – in theory – prevent this. These measures could be implemented in discrete logic or software, but a short circuit could still happen.

In power electronics we use the term 'shoot-through' for when both MOSFETs on the same side of the bridge are momentarily closed at the same time, even for just a few microseconds during the transition of the transistors changing state. The MOSFET switches' opening and closing can overlap due to delays caused by time constants based on gate impedance, gate capacitance, gate drive current and even down to the PCB design. To prevent this, a dead time is introduced so that the switching on of a transistor is delayed for a period of time longer than the switching off time. This means that during these dead times, all four MOSFETs will be switched off. However, solving the shoot-through problem leads to a new problem – with all switches open there is now nowhere for the motor current to flow

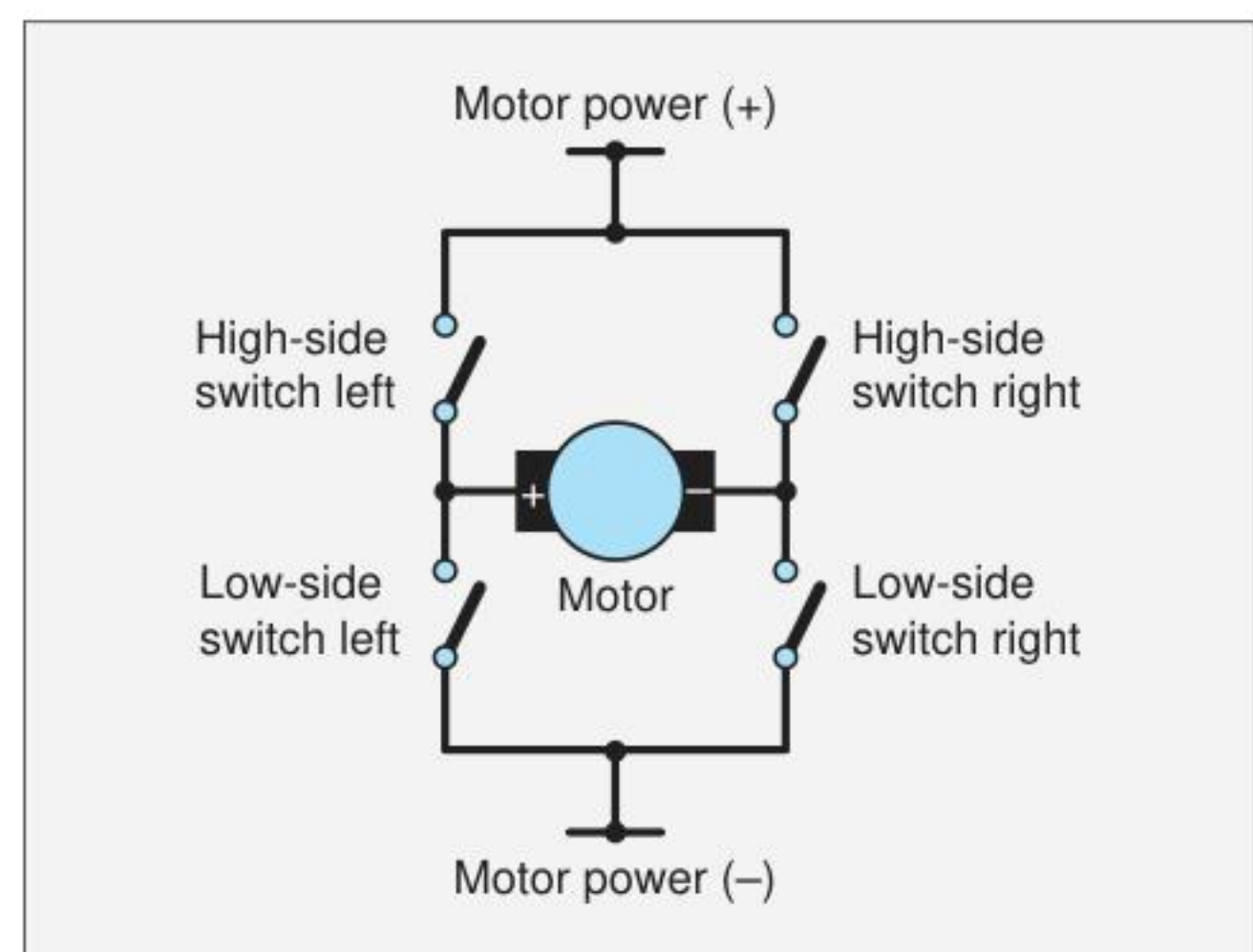


Fig.27. Four switches and a power source – the basis of an H-Bridge driver.

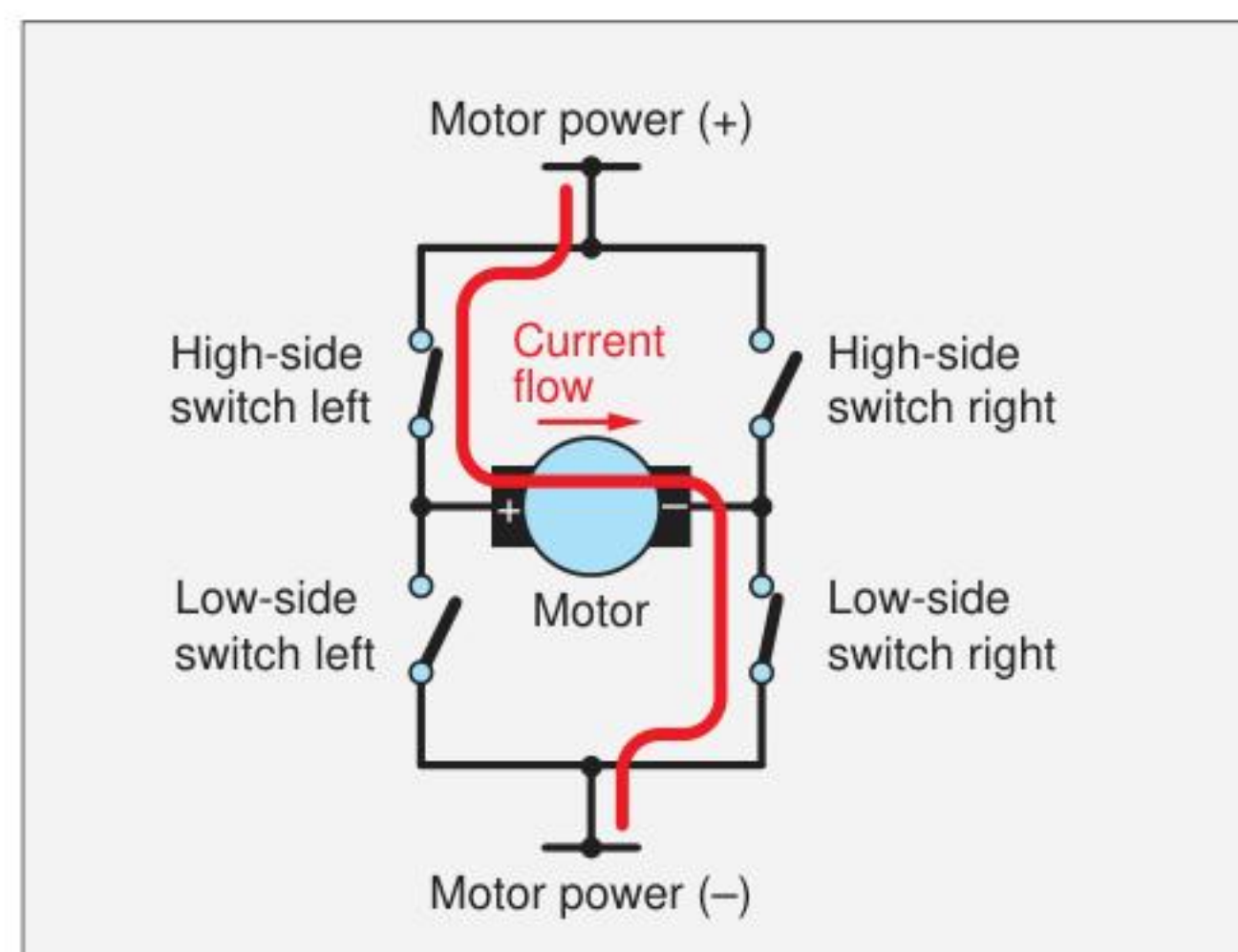


Fig.29. H-Bridge driver current flow for a specified motor direction.

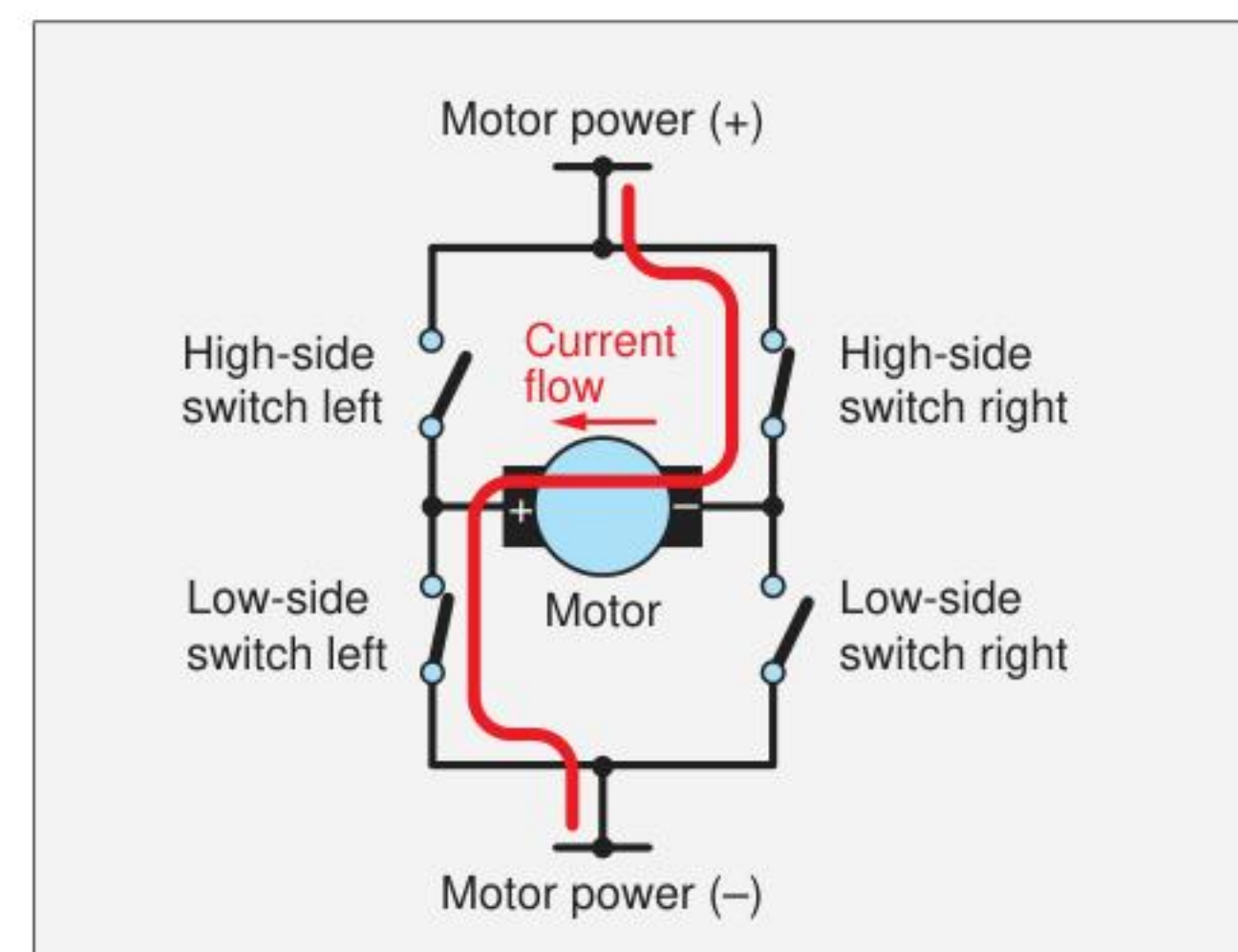


Fig.29. H-Bridge driver current direction for reversing the motor direction in Fig.29.

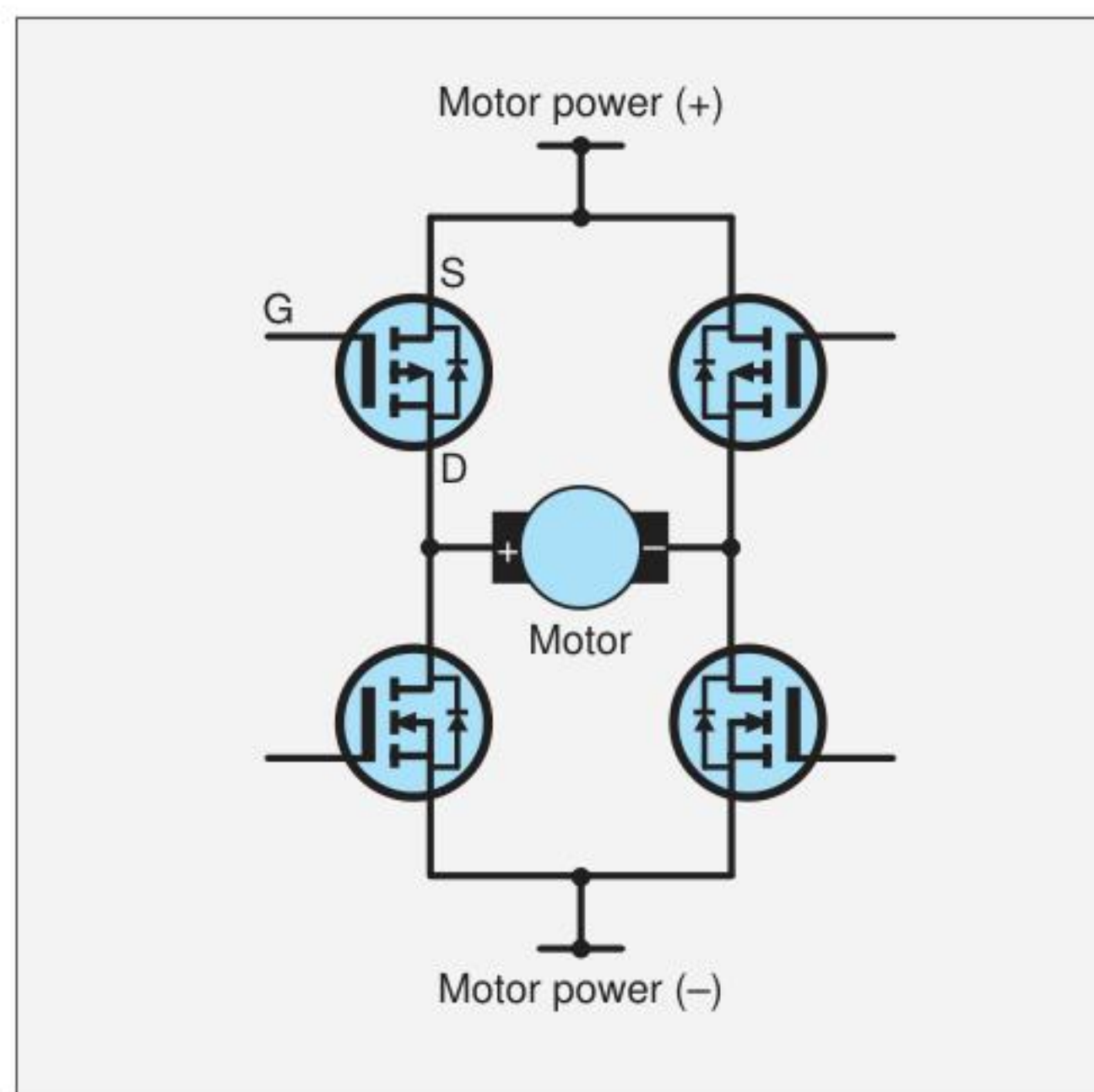


Fig.30. An H-bridge with MOSFET transistors – p-type high side and n-type low side. The diodes are *intrinsic*, they form part of the MOSFET transistor.

(the current resulting from the collapse of the magnetic field in the winding). This will cause the motor coil voltage to rapidly rise and potentially exceed the voltage rating of the MOSFETs and cause their failure – often a dramatic failure with smoke, even flames on high power MOSFETs. (I told you H-bridge design and operation is tricky!)

We need to ensure that at all times, a path exists for coil current flow and this is achieved with the ‘intrinsic’ (or ‘body’) diodes of the MOSFETs; or using external diodes if using bipolar junction transistors. It’s worth noting that MOSFET intrinsic diodes may not be fast enough or capable of dissipating sufficient energy from the winding currents during the dead time periods, so external diodes may also be required in MOSFET designs.

Even though there are other design considerations for the H-bridge, these are beyond this article. The purpose for dwelling on the pitfalls in H-bridge design is to demonstrate that despite their superficial simplicity they are not easy to build and operate reliably. The answer is to turn to proprietary stepper motor drivers – their key advantage is that all the timing complexity is taken care of inside the controller. This leaves you free to focus on the application for the motor driver and not be concerned about bridge designs.

Stepper driver main attributes

What features do we want from our stepper driver and why? Let’s examine three important requirements.

1. Microstepping

Our application may require very small motor steps for position control. Consider a 3D printer moving its extruder head in steps of fractions of a mm, perhaps 20µm or less. Assuming the stepper motor has 200 steps per revolution and is driving the extruder

head via an 18-tooth toothed GT2 belt (2mm pitch) then this arrangement gives a step distance of:

$$1 \text{ revolution} = 18 \text{ teeth} \times 2\text{mm} \\ = 36\text{mm of linear travel}$$

$$1 \text{ full step} = 36/200 \\ = 0.18\text{mm of linear travel}$$

If there was a way to make the step size smaller, then we could improve on the resolution. This is where an important stepper motor technique called ‘microstepping’ is used. Instead of a winding being fully energised or de-energised, intermediate values of current are applied. This allows the motor to take up positions between the teeth interval on the rotor. Half, quarter, 1/8, 1/16 steps are common place; some controllers can even go down to 1/256-steps, but at a cost in other performance areas. Using the above 3D printer example, 1/16 microstepping now gives a resolution of 0.18mm/16, which is 0.01125mm or 11.25µm.

To put this in terms of step angle; full-step rotation gives $360^\circ/200 = 1.8^\circ$ steps, but 1/16-stepping gives $360^\circ/(200 \times 16) = 3200$ steps per revolution, with a step angle of $1.8^\circ/16 = 0.1125^\circ$.

Many controllers use two pseudo-sinewaves to drive the windings using PWM (pulse-width modulation) techniques 90° out of phase with each other. As the current in one winding increases, the current in the other winding decreases. This provides a smoother motion compared to coarser discrete steps when using full, half or even quarter stepping. It is important to understand that a smaller step size isn’t always desirable or beneficial; for example:

- A point may be reached where the motor does not move because the step size is so small that the torque when stepping does not overcome the frictional losses in the motor. For example, 1/16 microstepping has around 10% of the holding torque available compared to full step, 1/256 microstepping is around just 0.6%. This loss of stepping will lead to a positional error with increased hysteresis.
- A smaller step size requires a higher clock signal frequency, which if derived from a microcontroller may be outside of its capability. For example, driving the above stepper motor at 600 RPM means a clock speed of $3200 \text{ steps} \times 600/60 = 32\text{kHz}$.

2. Current limiting

This is perhaps the biggest source of confusion for users new to bipolar stepper motor driving. Last month,

we described the speed limitation of a stepper motor due to the rate of change of the winding current caused by its inductance. Increasing the winding drive voltage allows for higher motor speeds and motor acceleration, but the current-limit action of the controller ensures the current rating of the motor is not exceeded.

This is important, so it’s worth expanding. A hybrid stepper motor might have a phase current of 1A at 3.7V, which you might reasonably think means 3.7V maximum. Well, that is true, but only after the winding current reaches a steady state. Following a step change in the winding current, the winding’s inductance opposes the current change. This opposition reduces until the rate of change becomes zero. If the stepper driver is operated at say 12V or even 24V, the stepper driver will apply as high a voltage as it can without exceeding the maximum current. This means shorter rates of change in the winding current, which means quicker accelerations and higher rotational speeds are possible. That is why 3D printers, which need fast accelerations, use power supply voltages far greater than the stepper motor nominal (steady-state) rating.

3. Minimum/maximum operating voltage and current

Matching the stepper motor driver to the motor is vital. The driver must be capable of delivering the full maximum phase current continuously if you want optimum motor torque performance. The minimum voltage rating of the controller needs to be below your chosen power supply and the maximum voltage rating above your power supply.

A4988 and DRV8825 stepper motor driver ICs

In the DIY 3D printer community, the A4988 and DRV8825 stepper driver chips from Allegro and Texas respectively are very well known, especially the A4988. They are only available in surface-mount packages, but this is not a limitation because they are readily available on compact, inexpensive breakout boards; for example, Fig.31. These breakout boards ‘break out’ the pins of a surface-mount chip like the A4988 into a pin spacing that is more aligned to through hole components and often suitable for direct use with breadboards. Additional components required by the chip may also appear on the breakout board.

The block diagram for the A4988 breakout board (Fig.32) is worth covering in some detail as it demonstrates how the increased complexity of bipolar drivers is taken care of in one, easy-to-use device.

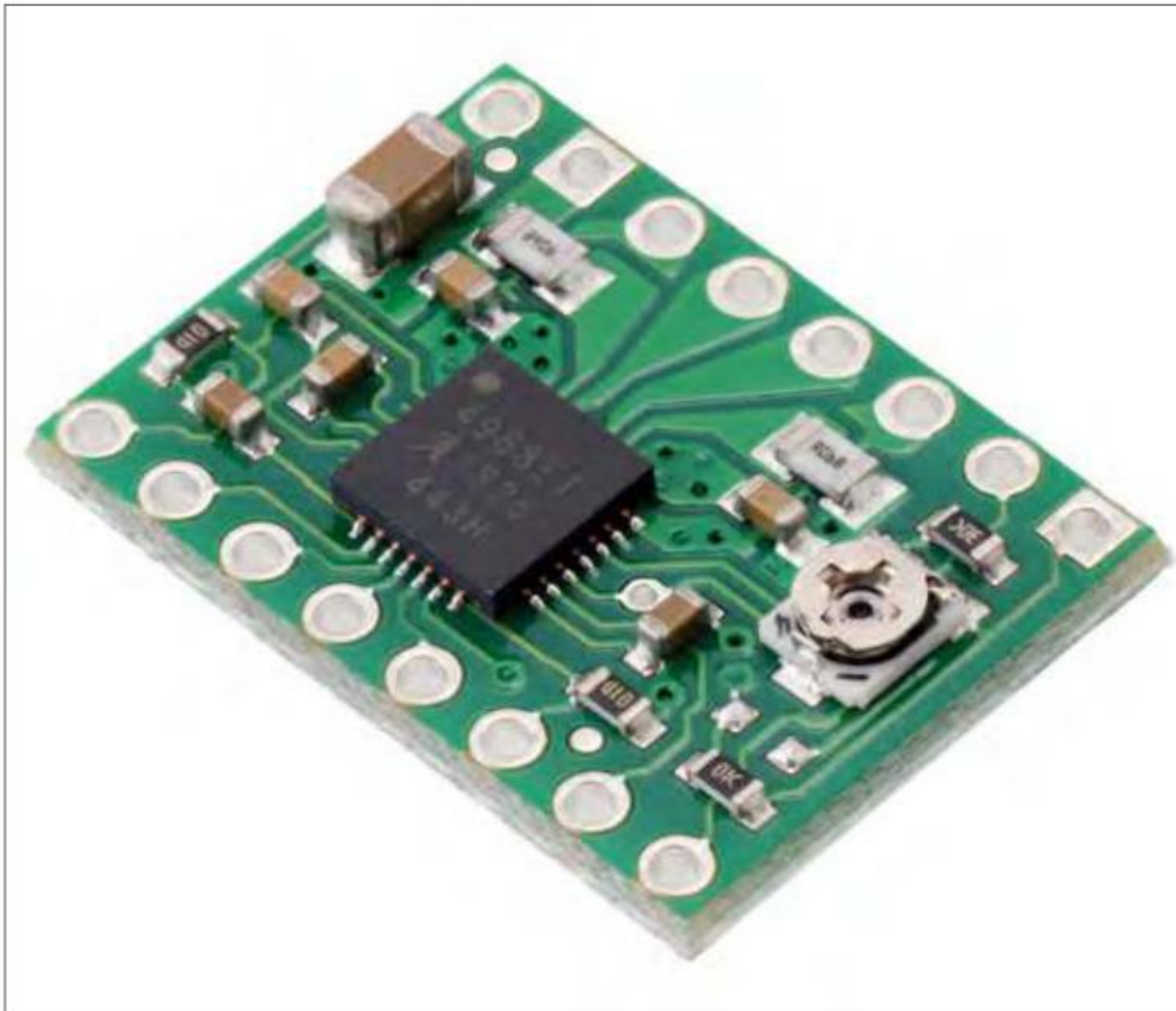


Fig.31. The Pololu A4988 breakout board.

In the block diagram you can see the two ‘H-bridges’, one for each winding. Note the high-side MOSFETs are also n-type, so the A4988 includes a charge pump for the higher-than-supply-voltage gate voltages. An external sense resistor is required for each bridge so that the winding current can be controlled as part of the A4988 current-limiting circuit. The control logic implements the various microstepping modes, interfacing and protection against ‘shoot-through’ by ensuring a fixed off time. So, while the A4988 chip implements all of the functionality we have so far come to

expect, it does need a few external components, which the breakout board takes care of.

Fig.33. Shows the schematic for the A4988 breakout board from the US-based manufacturer, Pololu – pololu.com

This board is available in the UK from technobotsonline.com for around £5. In addition to the external components specified in the A4988 datasheet, it also includes supply decoupling capacitors and a preset potentiometer (R9) for setting the current limit.

The A4988 breakout board is very easy to connect and use, the minimal wiring diagram shown in Fig.34 is largely self-explanatory, but briefly:

- Two power supplies are needed, one for the motor in the range 8-35V and a 5V supply for the logic, which can be derived from the microcontroller or other regulated supply. VMOT, the motor supply must have a 100µF capacitor across its supply lines, this is the only component you need to allow for; all others are on the breakout board
- The Enable input can be left unconnected, but if connected to +5V it will turn off the outputs to the motor but all remaining control logic remains active
- MS1, MS2 and MS3 – if left disconnected the controller will assume full-step mode. The various microstepping modes can be selected by taking appropriate connections to +5V, as shown in Table 7. Since the A4988 has internal pull-down resistors, pins can be left open if the low-state is required.

The reset and sleep connections are similar to the enable input in that they disable the outputs to the motor but additionally have further power saving functionality. The Allegro datasheet is recommended if you need to use these controls but if not, connecting these inputs together

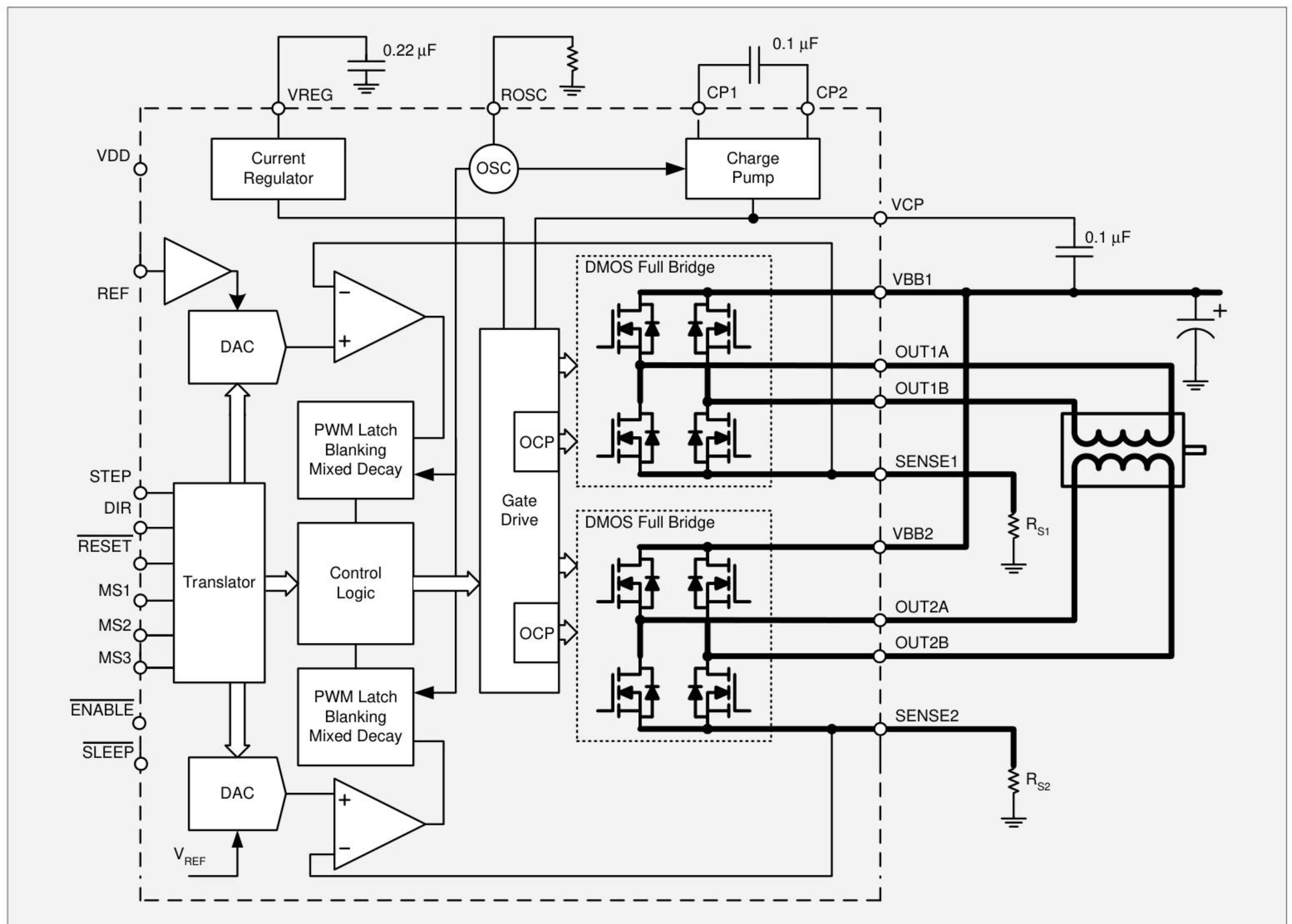


Fig.32. The A4988 block diagram from the Allegro datasheet.

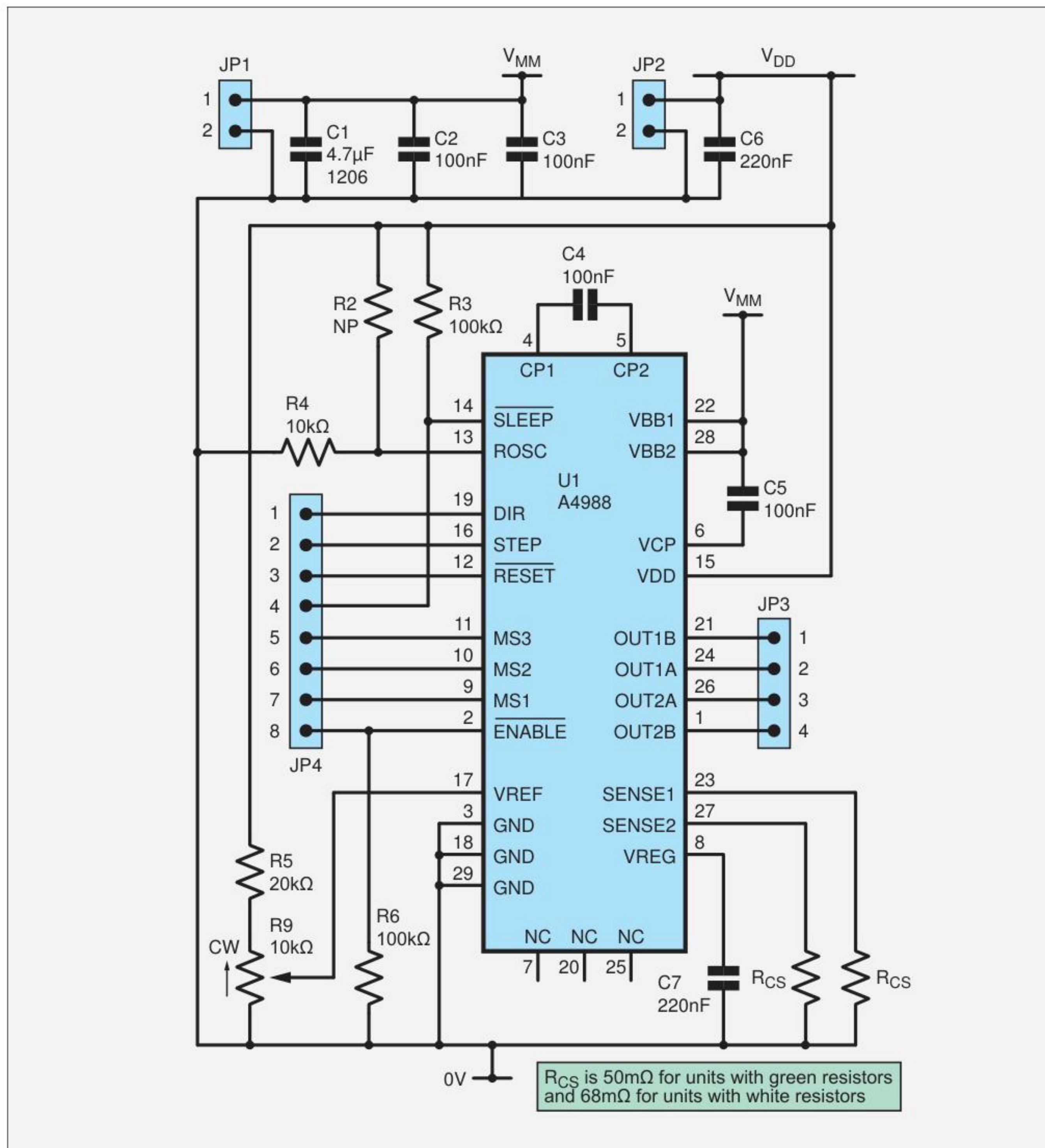


Fig.33. The Pololu circuit schematic for their A4988 based breakout board.

shares the same pull-up resistor on the breakout board.

Step and direction you will already be familiar with from previous articles in this series.

Finally, we come to the motor connections. As long as one winding is connected to 2A and 2B, and the other to 1A and 1B, your stepper motor will spin. You can change direction with the direction control pin, or simply swap 1A and 1B.

Setting the current limit

The final area to discuss this month is the all-important current limit. It is important that this is set correctly because otherwise the motor could either overheat or lack torque. There are a couple of ways to set the current limit, but the method I use has been proved to be successful.

From Fig.33 you can see that the trimmer potentiometer sets a voltage at the V_{REF} input on the A4988. First you calculate the V_{REF} voltage, and then you set it using the trimmer potentiometer *before* connecting the motor.

From the motor datasheet, the phase current will be specified. Let's assume that current is 1A and we wish to

Technobots

Supporting your projects since 2001

Mechanical

Bearings, Gears, Sprockets, Pulleys, Shaft, T-slot extrusion

Motors

Stepper motor drivers, DC controllers, Miniature and Geared Motors

Electronics

Passives, semiconductors, breakout boards, sensors, connectors and leads



MakerBeam

Pololu
Robotics & Electronics

FINGERTech
ROBOTICS

MFA/COMO DRILLS

MaxBotix

Order online at
www.technobotsonline.com

Follow us at
facebook.com/TechnobotsUK

Unit 29, Webb Ellis Business Park
Rugby, Warwickshire
CV21 2NP

Tel: 01788 877 400

current limit at this figure to achieve full performance. To do this we need to know the value of the current-sense resistors – all boards shipping from Pololu and Technobots have sense resistors of $R_{CS} = 0.068\Omega$. So, we use the following formula:

$$\begin{aligned} V_{REF} &= I_{MAX} \times 8 \times R_{CS} \\ &= 1 \times 8 \times 0.068 \\ &= 544mV \end{aligned}$$

Armed with a voltmeter, measure the voltage between the logic ground and the through hole PCB pad located between the trimmer and the A4988 chip – see Fig.34. Adjust the trimmer until you measure the calculated voltage. (I have also taken a small terminal screwdriver, attached the voltmeter positive lead to the screwdriver shank using a croc clip and the black lead to ground. You can then measure the voltage directly at the trimmer while adjusting it because the trimmer wiper is connected to the V_{REF} pin.)

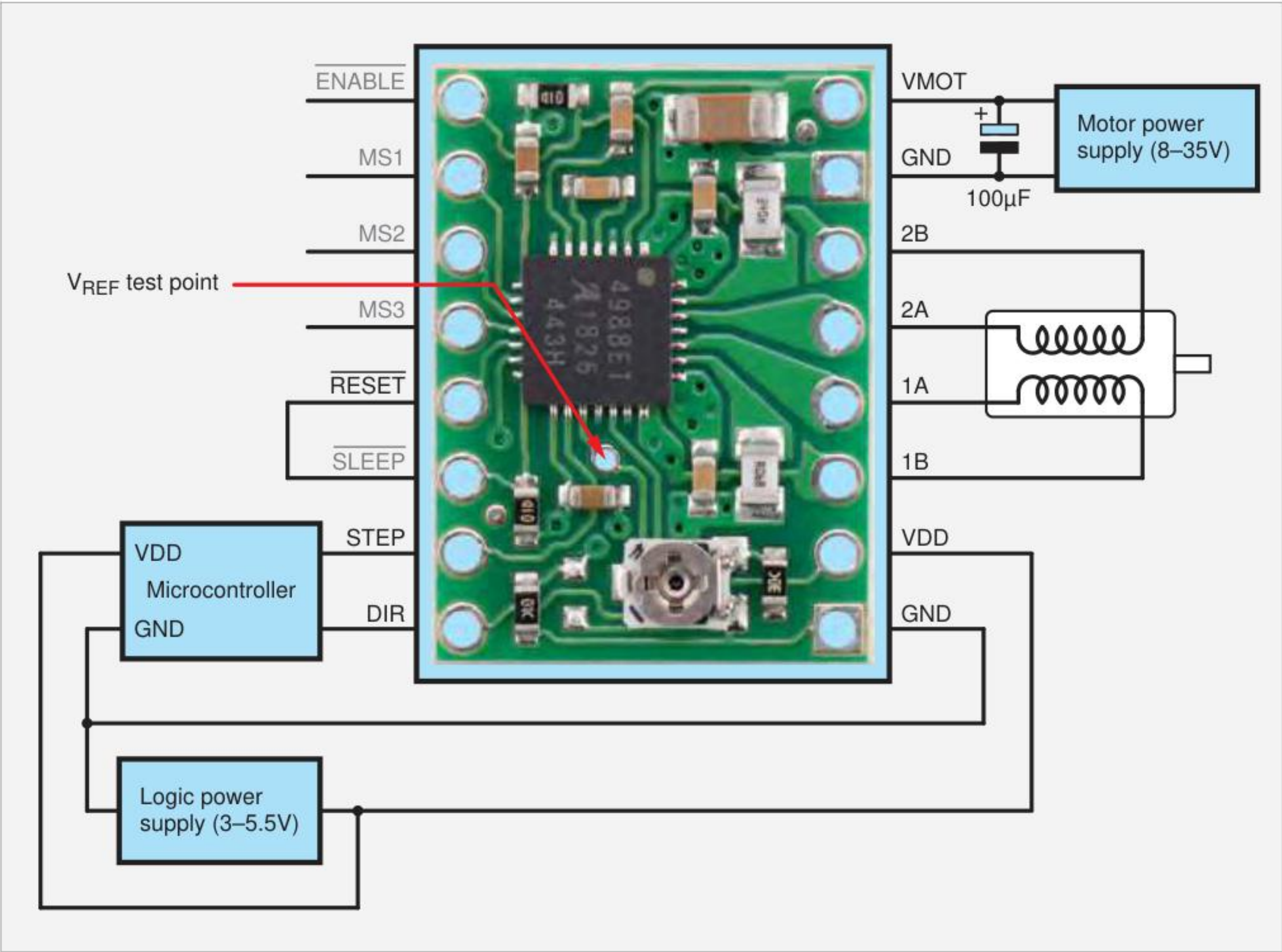
Next month

In Part 5, we will explore the features of many more bipolar stepper driver breakout boards because although the A4988 is great, just wait till you see what you can do with more advanced drivers!

MS1	MS2	MS3	Microstep resolution
Low	Low	Low	Full step
High	Low	Low	Half step
Low	High	Low	Quarter step
High	High	Low	Eighth step
High	High	High	Sixteenth step

Table 7: A4988 microstepping selection options using MS1, MS2 and MS3 control pins

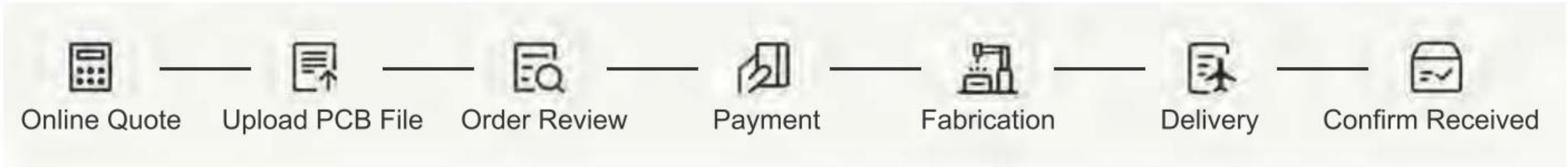
Fig.34. Wiring diagram for the A4988 breakout board from Pololu.



Win \$50 worth of PCBWay prototyping!

PCBWay is offering Practical Electronics readers the chance to win one of six \$50 credits to use with their world-leading PCB prototyping service.

Based in Shenzhen, China, PCBWay not only manufactures and assembles PCBs, but also offers its clients the most user-friendly, web-based interactive design system across the industry.



PCBWay's online web system lets you instantly get a quote for your PCB or PCBA order, and then track the online the status of its fabrication – all via your personal PCBWay account panel. After your PCBs have been dispatched, their shipping status can be checked on your order management page.

Our online facilities help keep everything under control, saving you time and energy all too often wasted in a frustrating loop of emails and phone calls.



How to enter

For your chance to win a \$50 free trial of PCBWay's prototyping service, just create your own PCBWay account at:

<http://bit.ly/pe-jan20-pcbway>

Closing date: The closing date for this offer is 1 January 2020

Max's Cool Beans

By Max the Magnificent



Don't just think about it, make it!

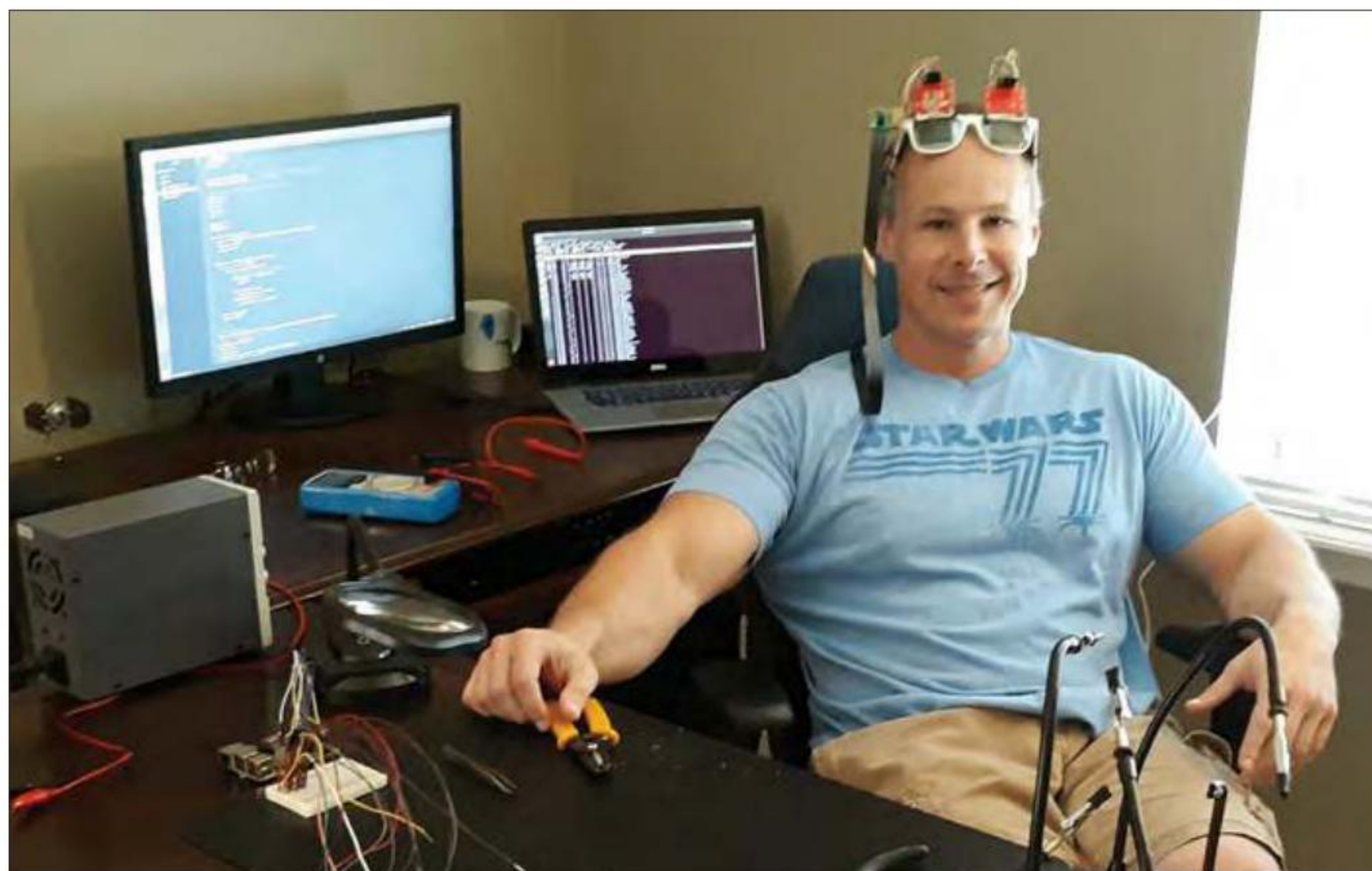


Fig.1. Nick Bild sporting his homemade *Light Brakes* smart glasses.



Fig.3. The *Vectron 64* LCD display.

As long as I can remember, I've been interested in understanding how things work, and I've dreamed of ways of improving devices and technologies. About a year ago, I had an epiphany, 'Why don't I just do it? Why can't I be the one to make something new and interesting?' I wasn't sure how to start, but there are so many educational resources available

As you may or may not know, I'm an engineer. I graduated in 1980 from Sheffield Hallam University with a BSc in Control Engineering – I think they call it 'Systems Engineering' now – which had a core of mathematics, along with mechanical, electrical, electronic and fluidic engineering.

My first job was as a member of a team designing central processing units (CPUs) for mainframe computers. I then drifted into electronic design automation (EDA), which encompassed the tools used to design and verify integrated circuits (ICs), printed circuit boards (PCBs) and embedded systems. Along the way, I've designed everything from silicon chips to circuit boards, and from brainwave amplifiers to steampunk *Prognostication Engines* (don't ask). Some of them even worked.

At some stage I drifted into writing, starting with magazine articles and working my way up to books – *Bebop to the Boolean Boogie* (<https://amzn.to/34oiP5D>) and *How Computers Do Math* (<https://amzn.to/2NwBbdJ>). From there, I took on contributing editor, editor, and editor-in-chief roles at a number of electronic magazines. Currently, I butter my bread with the proceeds from freelance technical writing.

One of the great things about being me (in addition to being outrageously handsome, a trend-setting leader of fashion and, of course, a global benchmark for modesty), is that I get to meet some amazing and interesting people, including members of the maker culture who like building weird and wonderful things just for the fun of it.

'A change is as good as a rest,' as they say, so I thought we'd have a bit of a change by introducing you to two of my maker chums. This month, we will look at the work of Nick Bild (Fig.1) in the US; and in my next *Cool Beans* I'll introduce you to Richard Grafton in the UK. The following is in Nick's own words.

Say hello to Nick Bild

All of my formal education and experience is on the software side. I have degrees in computer science and bioinformatics; also, I've spent 15 years as a software engineer.

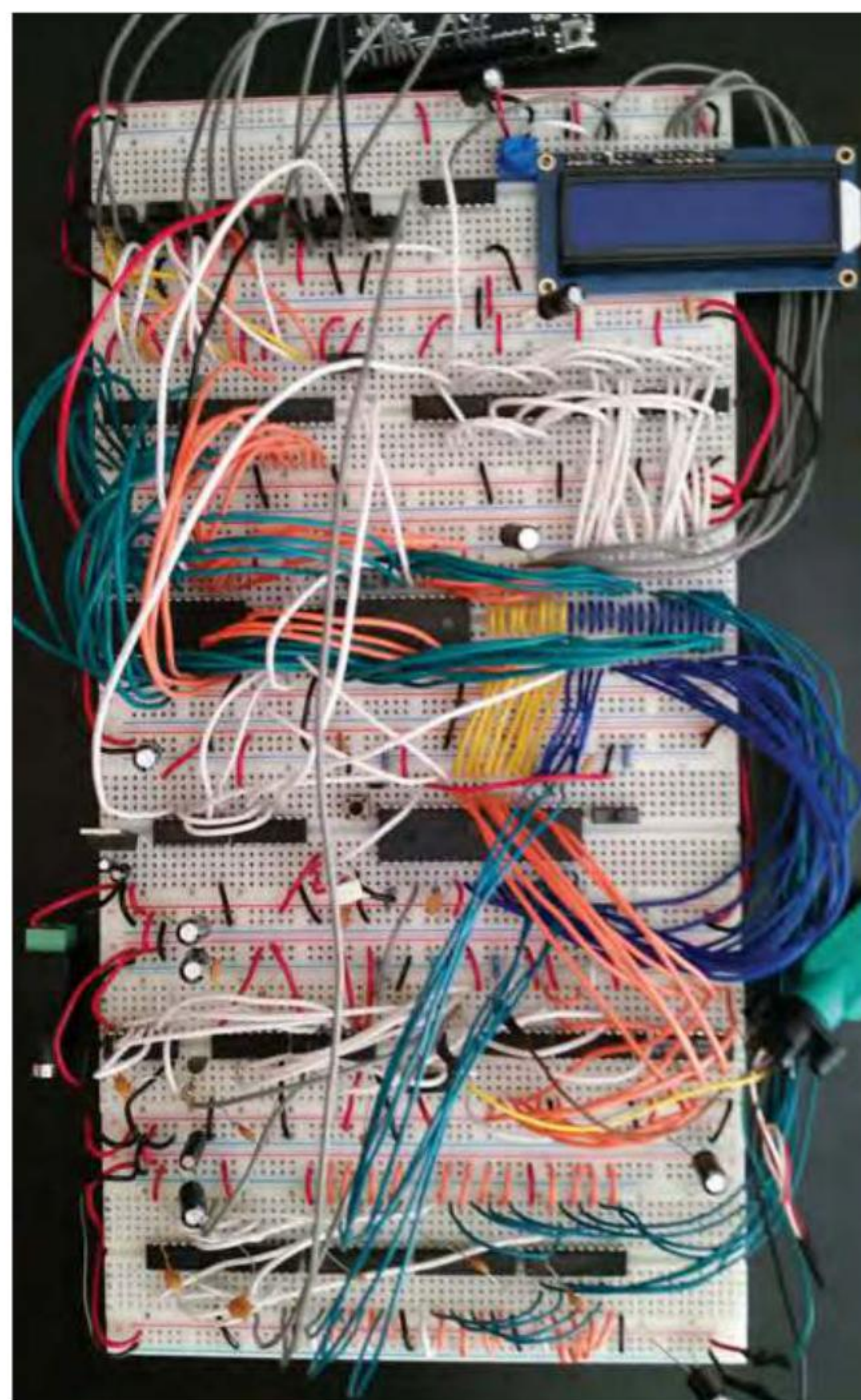


Fig.2. The *Vectron 64* breadboard computer.

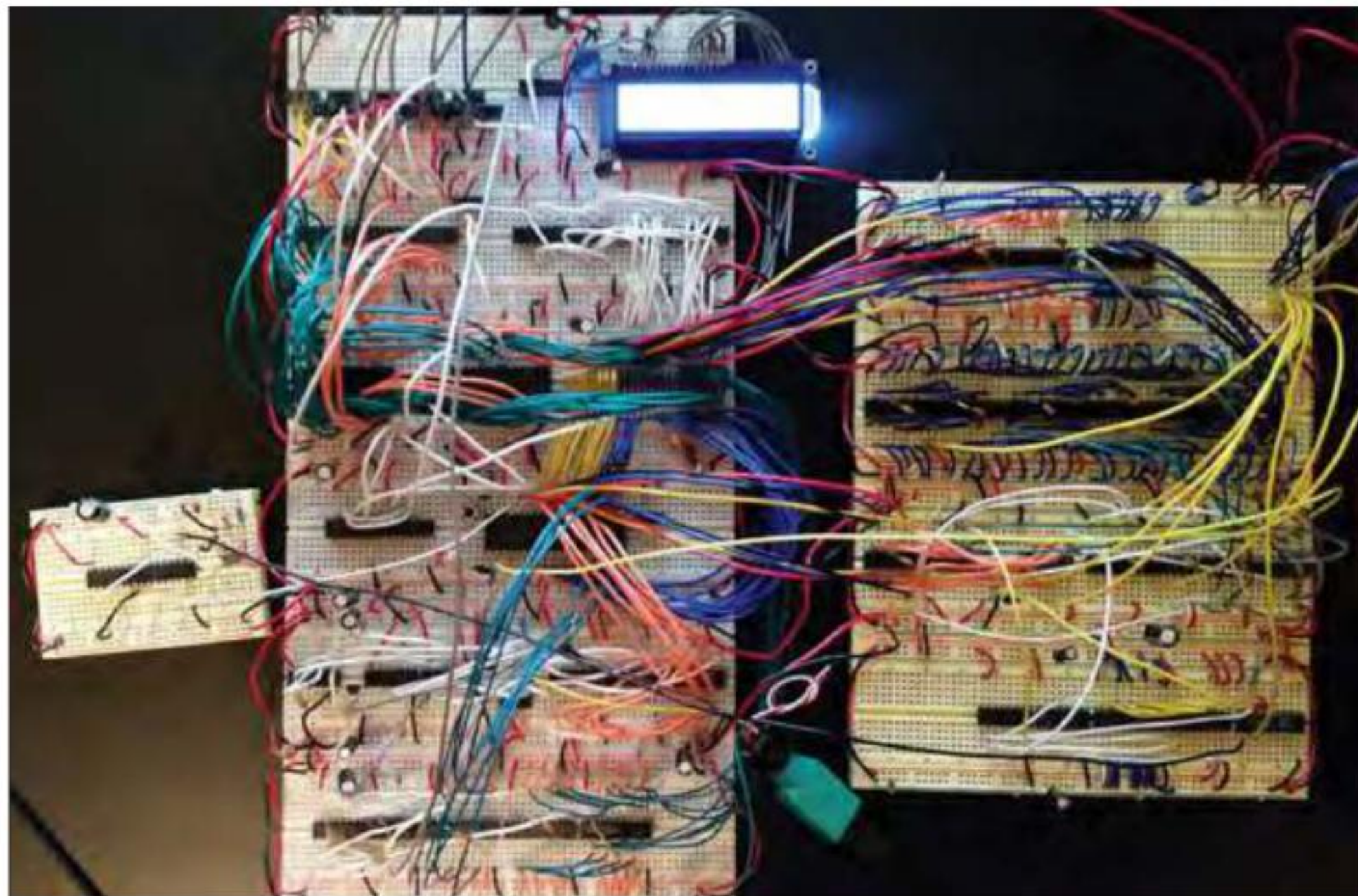


Fig.4. The Vectron 64 (left) augmented with VR capability (right).

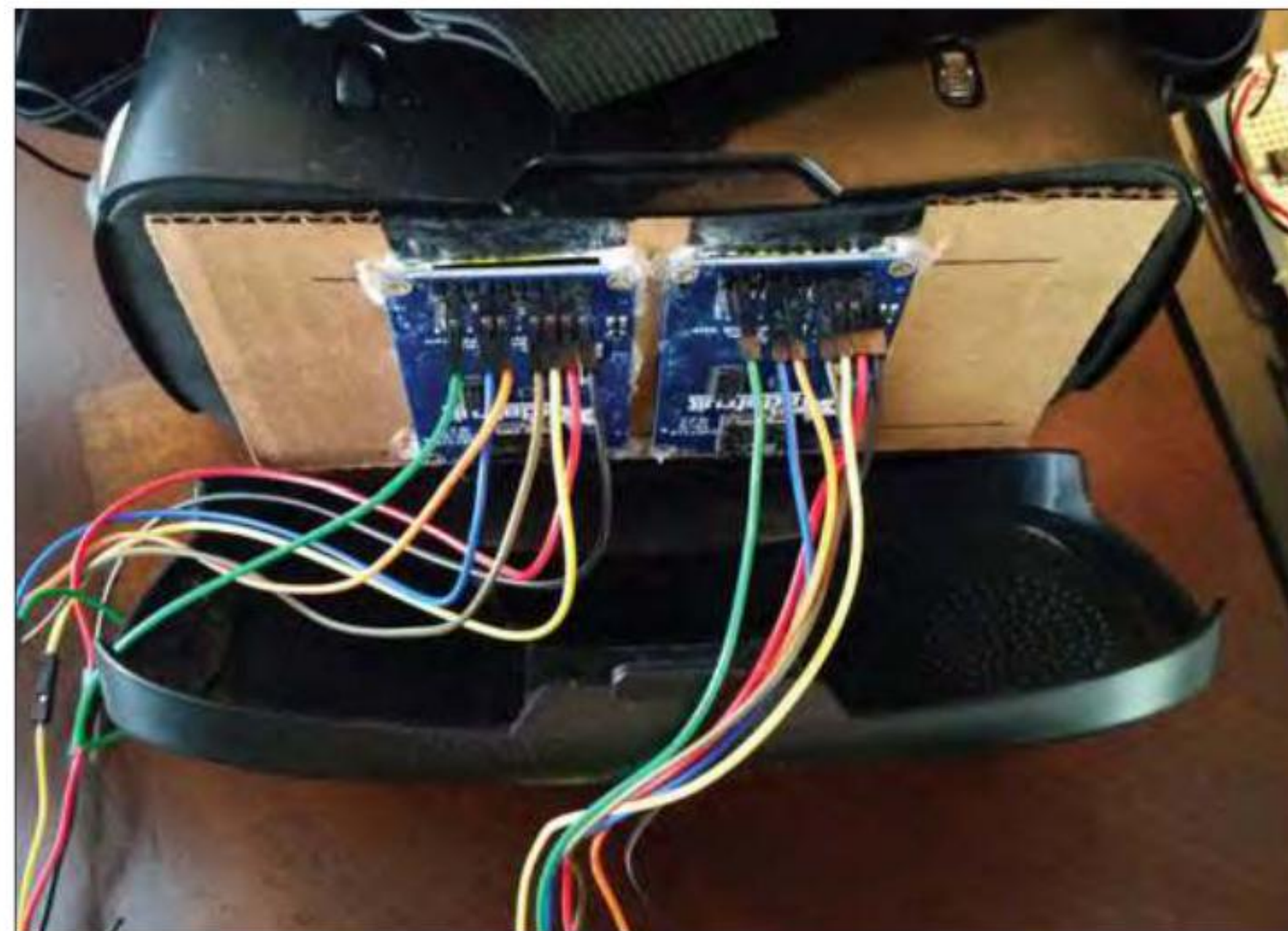


Fig. 5. The VR headset opened up (the LCD screens and lenses are behind the circuit boards).

on the Internet that I was able to break down my ideas into a series of steps, do some research, and devise methods to complete each step. Completing my first significant project (the *Vectron 64*) got me hooked.

Vectron 64

The *Vectron 64* (Fig.2) is a programmable 6502-CPU-based breadboard computer that I designed and built. The 6502 (or minor variants) powered classic computers such as the Commodore 64, BBC A/B and Apple II. I took on this project to teach myself about electrical engineering. I began the project knowing almost nothing, literally teaching myself about the difference between voltage and current as a first step. A few months later, I had a custom-built computer with an 8-bit CPU, 32K of RAM, and 32K of ROM, along with a PS/2 keyboard and an LCD display for input and output (Fig.3). All of the files associated with this project are available on GitHub (<http://bit.ly/34pf4g2>).

The Vectron VR

For my next project, I wanted to make my *Vectron 64* do something unexpected or improbable. After some thought, I decided that a virtual reality (VR) headset for a computer built with 1970s technology would fit the bill nicely – if I could make such a thing work.

I was able to drive two 320 × 240 LCD screens independently by using a few hardware tricks and writing some tight assembly code. I offset sprites by a few pixels between the two screens to create stereoscopic 3D visual effects, then I mounted the screens in a VR headset meant for use with a smartphone. I wrote a 3D game called *Asteroids VR* to demonstrate the system.

I learnt a lot from building the computer, which helped me in getting this project off the ground. I did run into lots of issues, especially with the timing of digital signals, which required me

to improve my debugging skills and get acquainted with a logic analyser. (If only I'd had a logic analyser when I was adding the keyboard controller to the *Vectron 64*, I could have shaved weeks off the project! Oh well, hack and learn!) Once again, all of the files associated with this project are available on GitHub (<http://bit.ly/2pBCbVY>).

Doom Air

Having spent several months working on decades-old technology, I wanted to work on something more current. I decided it would be valuable to teach myself about artificial intelligence (AI) [Editor's note: see Max's column *What the FAQ are AI, ANNs, ML, DL, and DNNs?* (<http://bit.ly/2JHLBWQ>)], so I came up with a fun project idea to play the game *DOOM* in a new way.

I trained a convolutional neural network (CNN) to recognise a number of positions that I put my body into. Next, I projected *DOOM* onto the wall in front of me and pointed a camera in my direction (Fig.6.). The CNN recognised what gesture I was making, using that information to control the game.

By standing in front of the screen, and making natural movements to control the action, it created a very immersive experience. As usual, all of the files associated with this project are available on GitHub (<http://bit.ly/2oEVPjF>).

ShAIdes

My next build was a set of AI-powered smart glasses that I call *ShAIdes*, which have been the most popular of all my projects thus far.

I have an interest in augmented reality (AR) [Editor's note: see also Max's column *What the FAQ are VR, MR, AR, DR, and HR?* (<http://bit.ly/2WyN4Ed>)], so I was thinking about current devices. It occurred to me that most people probably wouldn't consider using current devices because they simply don't want a screen in front of them at all

times. That gave me the idea to take AR literally. That is, augment the real world (rather than an overlaid display) by simply gesturing to the things you are looking at. No display would be needed; the real world is your display.

I built a device that allows me to toggle my lights on or off, simply by looking at them and waving my hand. I can also play music on my smart speakers in the same way. Using the methods I developed, a great many network-connected devices can be controlled with a simple gesture. As always, all of the files associated with this project are available on GitHub (<http://bit.ly/2r2FfLf>).

Light Brakes

My most recent build is my *Light Brakes*, which are smart glasses that selectively filter out excessively bright spots (eg, the Sun, headlights, glare) from a wearer's field of vision in real time. The remainder of the view is unobstructed and transparent.

I accomplished this by attaching a camera to a pair of glasses to spatially locate areas of excessive brightness. I



Fig.6. This Jetson Nano was trained to recognise my body position.



Fig.7. Using *ShAldes* glasses, the real world is your display.

then fitted a disassembled graphic LCD screen (with the backing removed to render it transparent) over each lens of the glasses. I translate the bright spots into coordinates on the LCDs and draw pixels to darken those regions.

I came up with the idea on one of my commutes to work. I drive east in the morning, and west in the evening – always towards the Sun. I firmly believe that looking for solutions to real problems that you have is one of the best ways to come up with ideas for impactful projects. Once again, all of the files associated with this project are available on GitHub (<http://bit.ly/2PGk3ov>).

Back to me (Max)

Well, I don't know about you, but I'm tremendously impressed. I'm blown away by the fact that when Nick started his first project – the *Vectron 64*, which is essentially a 6502 computer on breadboards – he knew almost nothing, literally having to learn the difference between voltage and current as a first step. I hope his creativity and success will inspire *PE* readers to start their own projects.

In my next column, we'll look at the work of England's very own maker, Richard Grafton. In the meantime, I'd love to hear your comments and questions. Until next time, have a good one!

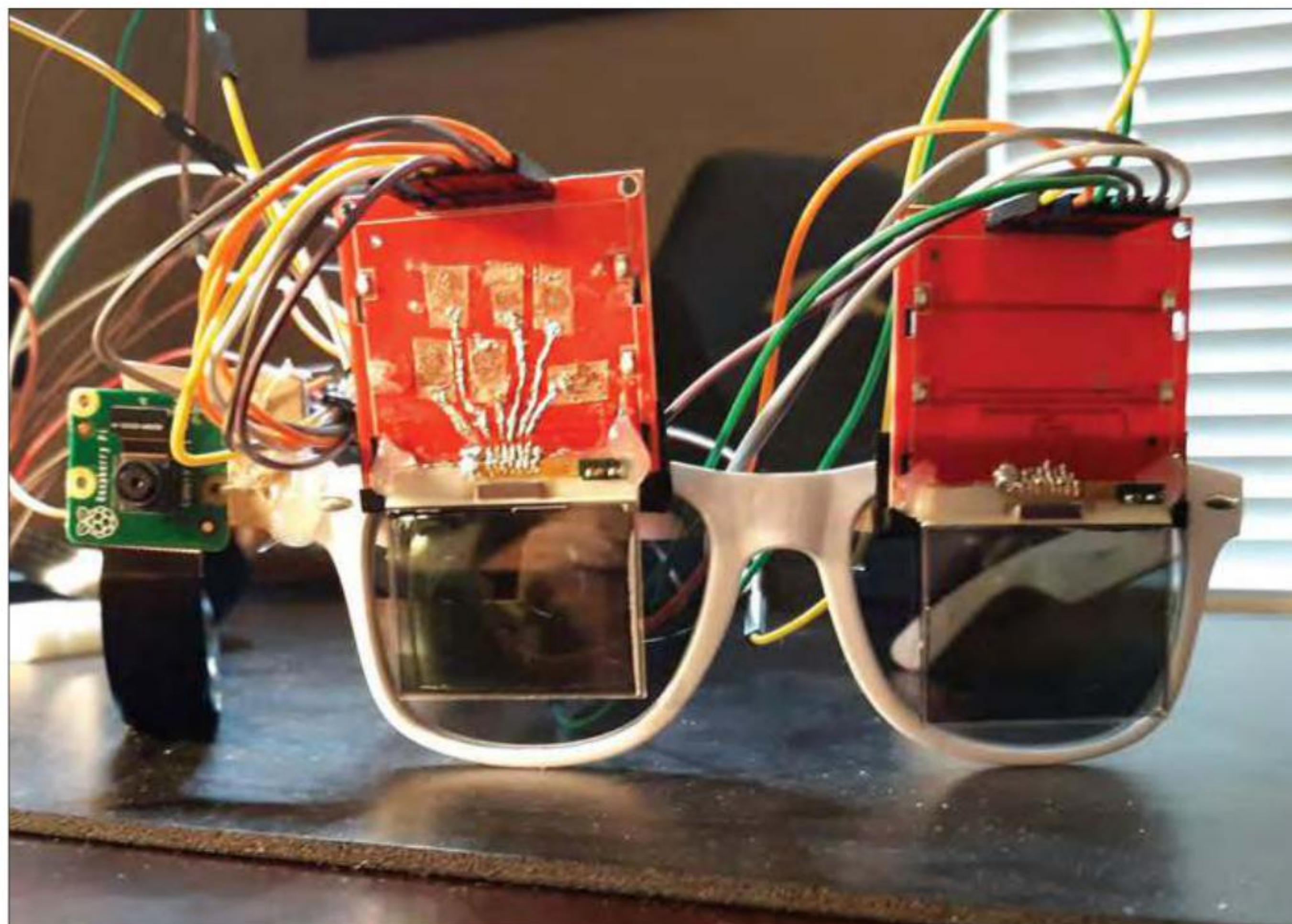


Fig.8. *Light Brakes* are smart glasses that selectively filter out bright spots.



Cool bean Max Maxfield (Hawaiian shirt, on the right) is emperor of all he surveys at **CliveMaxfield.com** – the go-to site for the latest and greatest in technological geekdom.

Comments or questions? Email Max at: max@CliveMaxfield.com

PoLabs

For more information or software download please visit 

www.poscope.com/epe

PoKeys

Connect, Control



- USB
- Ethernet
- Web server
- Modbus
- CNC (Mach3/4)
- IO
- PWM
- Encoders
- LCD
- Analog inputs
- Compact PLC

Stepper motor drivers

Drive



- up to 256 microsteps
- 50 V / 6 A
- USB configuration
- Isolated
- up to 32 microsteps
- 30 V / 2.5 A

PoScope Mega1+ PoScope Mega50

Measure



- up to 50MS/s
- resolution up to 12bit
- Lowest power consumption
- Smallest and lightest
- 7 in 1: Oscilloscope, FFT, X/Y, Recorder, Logic Analyzer, Protocol decoder, Signal generator

Circuit Surgery

Regular clinic by Ian Bell

Logic levels – Part 1

This month's **Circuit Surgery** topic was suggested by *PE*'s editor, Matt Pulzer – logic levels and related issues with interfacing devices. The term 'logic level' refers to the voltages used to represent logical 1s and 0s in digital circuits. We usually design logic circuits by considering just the 1s and 0s of data and control signals in combination or sequences. However, the logic gates and flip-flops from which these circuits are built are in turn implemented using transistors – and these circuits have analogue properties if you look at the relationship between their inputs, output and supply voltages at a sufficient level of detail.

Most of the time it is not necessary to worry about this – it is possible to design processors with ten thousand million transistors in them because we can abstract away the analogue aspects of the circuits and just deal with a digital representation of the signals. However, sometimes we do have to consider the actual voltages and other 'analogue' aspects of logic circuits, this includes when interfacing devices from different technology families; operating on different supply voltages; or when considering the effects of electrical disturbance on circuit operation. To highlight potential differences between circuits we will look briefly at two key logic types – CMOS and TTL – then look at logic voltages in detail and a quick example of practical interfacing.

In this article we use the terms 'logic technologies' and 'logic families'.

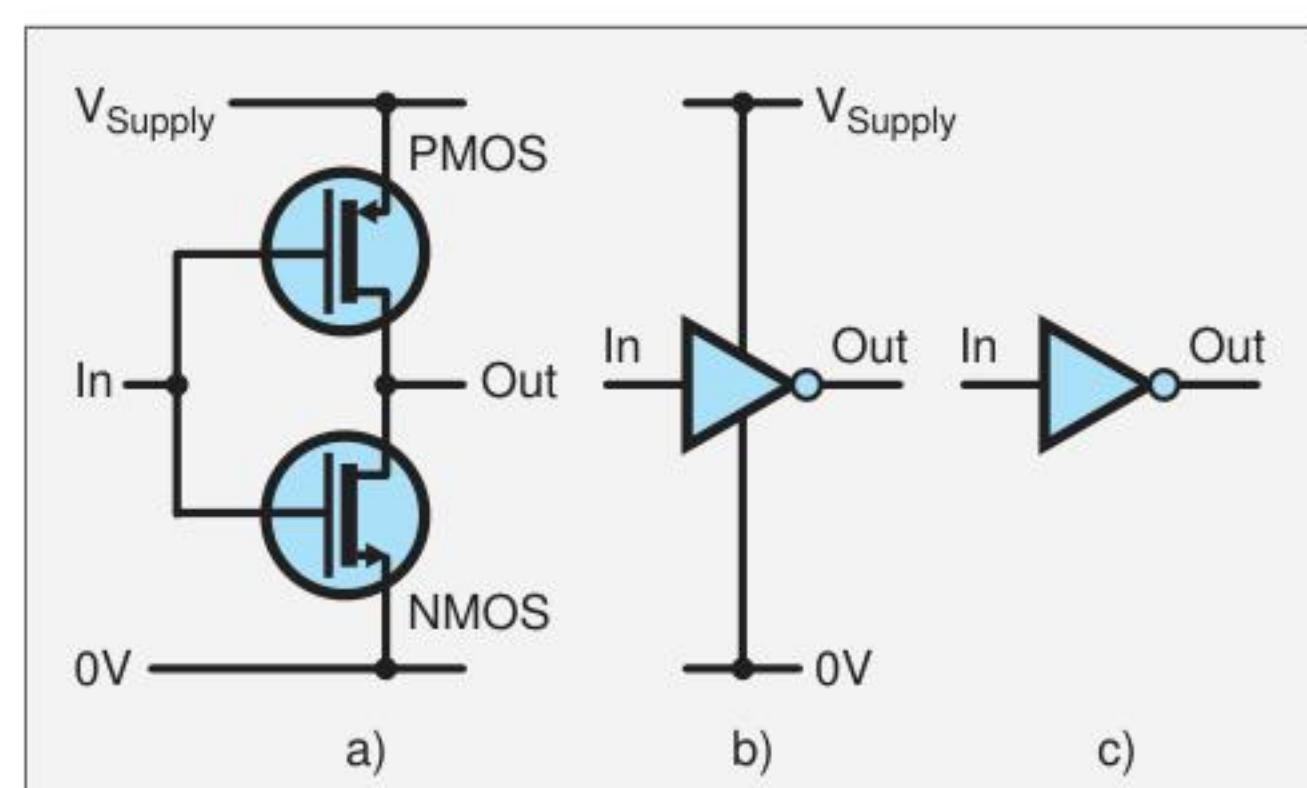


Fig.1. CMOS NOT gate: a) Transistor circuit for CMOS NOT gate, b) NOT gate using gate symbol and showing supplies, c) NOT gate using gate symbol without showing supplies.

'Technologies' refers to the fabrication processes used to produce all types of integrated circuits, including, for example, high-end microprocessors. The technology determines the type and size of transistors used and many characteristics such as speed and power consumption. 'Logic family' refers more specifically to the various ranges of basic logic devices (such as gates, flip-flops, registers and counters) produced by manufacturers such as Texas Instruments. Each logic family is implemented in a particular technology and uses a particular logicgate design approach at transistor level.

Defining levels

Logical and computing systems, from the simplest functions through to advanced arithmetical and algorithmic processes can be designed and analysed in terms of mathematical and abstract representations based on Boolean algebra, binary numbers and various coding systems. To actually build circuits to implement these ideas we have to decide how to represent 1 and 0 in a real electronic circuit. We could use two voltages, say +5V for logic 1 and 0V for logic 0, but this is arbitrary. It could be -2V for 0 and +2V for 1, or 0V for 1 and +5V for 0. In general, if the more positive voltage is used for 1 we refer to this as positive logic and if the more negative voltage is used for 1 we have negative logic. Things can get confusing; for example, the same physical circuit can act as a NAND gate using positive logic, but a NOR gate using negative logic. Fortunately, the vast majority of circuits that you are likely to encounter will use positive logic.

If we select 5V for logic 1 and 0V for logic 0 then what does 4.9V mean? In real circuits we have to define a range of voltages that represent a valid logic level, say 0V to 2V for 0, and 3V to 5V for 1. We need to do this because we cannot build circuits which handle precisely fixed voltages under varying conditions of loading, supplies and temperature, and with variations in individual manufactured devices.

The voltages used to represent 1s and 0s will depend on the supply voltage used to power the circuit. Usually, one supply is ground (0V) and the other is at a positive voltage with respect to ground (the supply voltage). For positive logic, the ideal logic 1 will be equal to the supply voltage and the ideal logic 0 will be 0V. As just discussed, a range must be used, so the logic 0 range will be from 0V to a relatively small positive voltage, and logic 1 will be from a larger positive voltage up to the supply voltage. We will look at this in a little more detail shortly.

Logic supplies

Having mentioned power supplies, it is worth pointing out that logic circuit schematics are typically drawn without power supply connections being shown. This is simply because if we drew all the supply wiring it would clutter the diagram up with extra lines which would make it more difficult to see the structure of the logic. Fig.1 shows the schematic of a NOT gate implemented in its most basic form in CMOS technology (more on this soon) – we usually include the supplies when looking at logic circuits at this detailed level. The figure also shows the same circuit drawn using the familiar NOT gate symbol, both with and without the supplies – the latter clearly focusing on just the logical function.

The actual voltages used to power digital circuits have become more diverse as technology has progressed. There was a time when many digital ICs either used exclusively 5V supplies or were happy working at 5V (within a larger possible range). Since then, there has been a downward trend in supply voltages, with typical values being 3.3V, 2.5V, 1.8V, 1.5V, 1.2V and 0.8V. Thus, it is not uncommon to find that two key chips or modules that provide the right functions for a design require different supply voltages and hence have potentially incompatible logic levels.

The downward trend and associated diversification of supply voltages of available chips is related to the physics

of scaling transistors down to small dimensions. This is the driving force for our increasingly powerfully digital technology, which is described by Moore's law (the observation that the number of transistors in a dense integrated circuit doubles about every two years). The impact of scaling transistor sizes on the characteristics of IC technology is complex. The reduction in supply voltages is related to power dissipation, leakage currents (which affect standby power) and electric field strengths within devices. This downward trend in supply voltages more or less followed the trend in transistor size reduction from the 1990s to the mid-2000s, but then levelled off, while transistors continued to get smaller, more or less in accordance with Moore's law.

MOSFET-based logic

The NOT gate (inverter) in Fig.1 uses CMOS technology. CMOS stands for 'complementary metal-oxide semiconductor', where 'complementary' refers to the fact that the circuit implements both types of MOSFET transistors (PMOS and NMOS). CMOS is the most commonly used technology for digital circuits today. At the most basic level, the MOS transistors in CMOS logic act as electrically controlled switches, with the gate voltage controlling conduction between the source and drain. PMOS and NMOS transistors require opposite gate-source polarities to switch on. In the NOT gate shown in Fig.1, a logic 1 input (close to V_{Supply}) will switch the NMOS transistor on and the PMOS off, this connects the output to ground (0V) producing a logic 0 at the output – as required for logical NOT. For logic 0 (0V) input the opposite conditions occurs (PMOS on, NMOS off) so the output is connected to the supply producing a logic 1 output as required.

CMOS was not the first logic technology using MOS transistors. Developments started in the early 1960s with the first ICs being PMOS-only because NMOS transistors were too difficult to manufacture at the time. By the mid to late 1960s NMOS-only and CMOS were also available and RCA introduced the well-known 4000 series of devices. NMOS was faster than PMOS, and as manufacturing techniques improved, by the late 1970s it came to dominate MOS-based logic. With further advances, CMOS which has similar speed, but much lower power than NMOS, took over as the most common logic technology (eg, for processors) during the 1980s. Numerous CMOS logic families have continued to be released over the years, with names such as

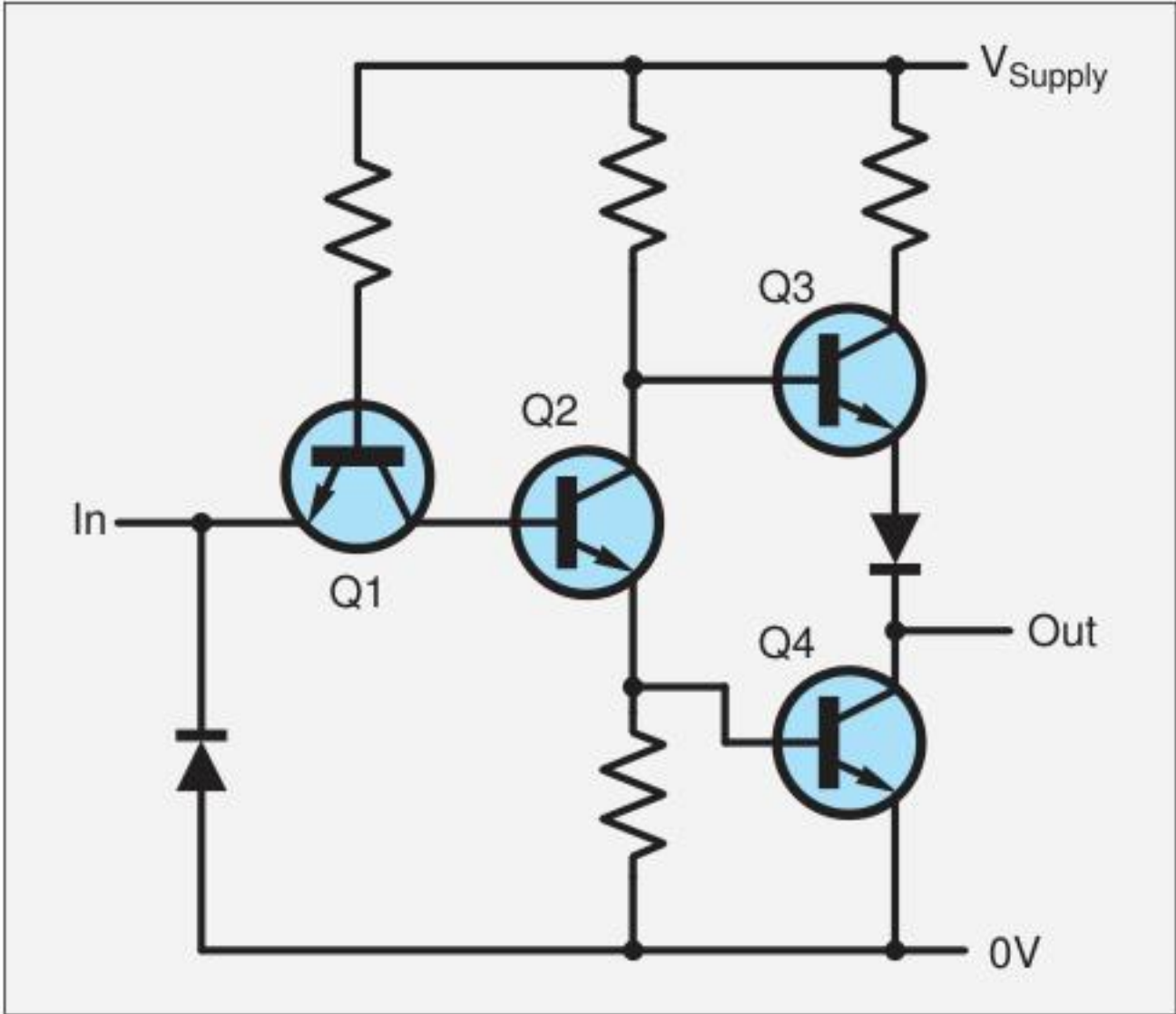


Fig.2. TTL NOT gate (inverter) schematic.

High-Speed CMOS (HC/HCT), Advanced CMOS (AC/ACT) and Advanced Ultra-Low-Voltage CMOS (AUC). The various CMOS families operate on different supply voltages and can operate at different maximum speeds.

Bipolar transistor-based logic

CMOS is not the only logic technology, particularly when you look at the overall history of development of integrated digital electronics. As with MOS-based logic, developments get going in the early 1960s with various bipolar junction transistor-based ICs:

- Resistor Transistor Logic (RTL)
- Diode Transistor Logic (DTL)
- Transistor Transistor Logic (TTL).

TTL became the dominant version of these technologies, particularly with the release of the 7400 series of devices by Texas Instruments in the mid-1960s. TTL continued to develop, with the use of Schottky transistors and other developments leading to faster, lower-power devices. Different families of devices were released over the years, in particular: Low-power Schottky (LS), Advanced Low-power Schottky (ALS), and FAST (F) – Fairchild Advanced Schottky TTL.

Although well known, TTL is now a somewhat obsolete technology, with use declining since the late 1990s. However, devices are still available and 'TTL compatible' logic levels are commonly used. Some logic families have two versions (eg, AC and ACT) with the 'T' version indicating TTL-compatible levels. CMOS and TTL are not the only major types of logic. There are also ICs which combine the two types of transistor in what is known as 'BiCMOS technology'.

Fig.2 shows the schematic of a basic TTL inverter. Like the CMOS circuit in Fig.1, this can also be represented as a NOT gate symbol when drawing a gate-level schematic. The circuit has three stages: current steering (Q1), phase splitting (Q2) and a 'totem-pole' output driver (Q3 and Q4). The input controls

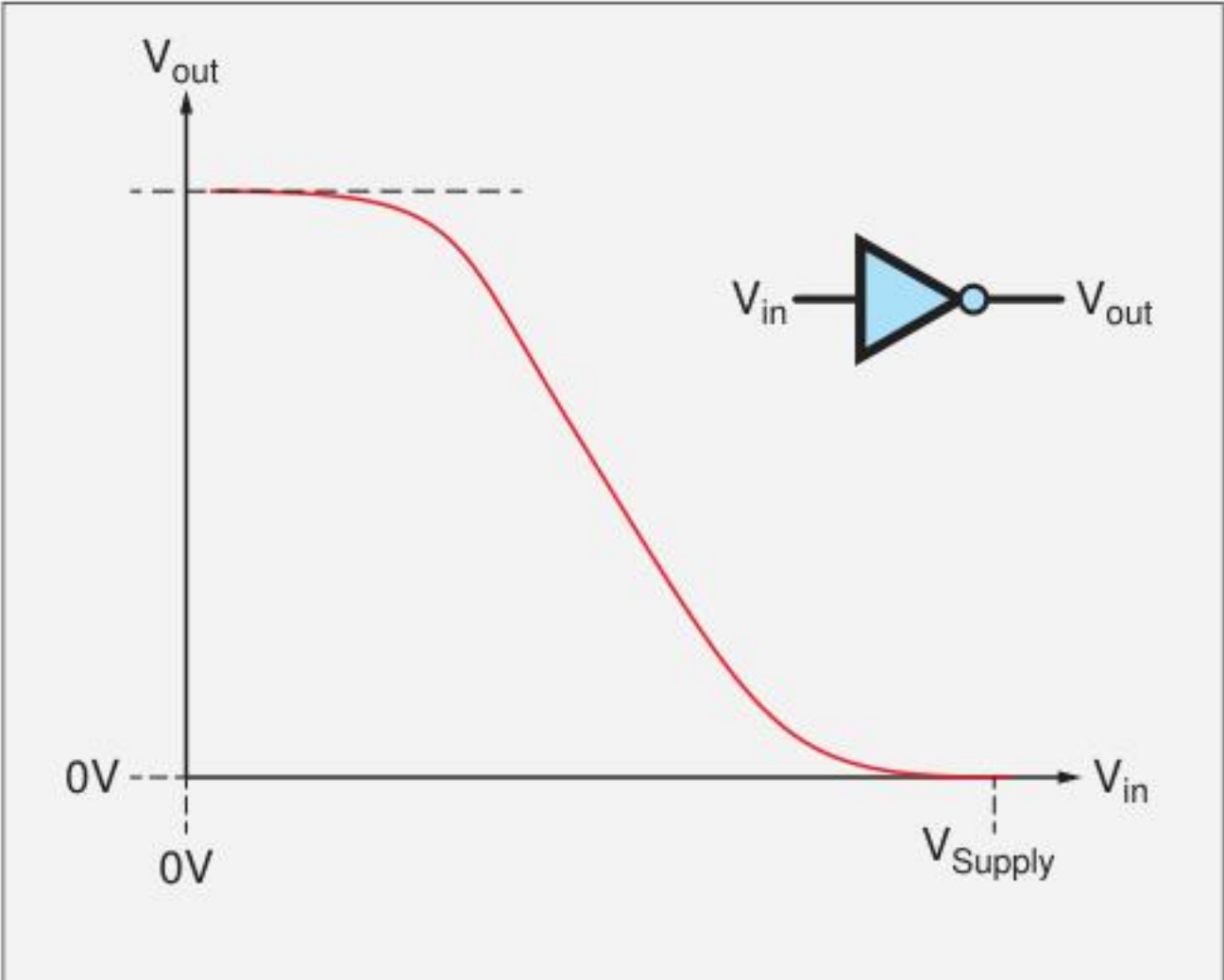


Fig.3. Sketch of an inverter transfer characteristic.

the direction of current in Q2's base via Q1. Q2 in turn drives the bases of Q3 and Q4 in opposite directions. This is required because, unlike the CMOS circuit, the two output transistors are of the same type (NPN) so their bases need to be driven differently to switch one off while the other is on.

For the circuit in Fig.2, with 0V at the input, Q1 causes current to flow away from Q2's base, switching Q2 off, resulting in a voltage close to the supply at Q3's base and 0V at Q4's base. This switches Q3 on and Q4 off, giving a logic 1 at the output. If the input is not pulled to a low voltage, Q1 provides current into Q2's base, switching it on. This pulls down the voltage at Q4's base low enough to switch it off, but the voltage drop across the resistor at Q2's emitter is sufficient to switch on Q4, giving a logic 0 at the output.

Logic voltages

The different transistor types, NOT gate schematics (Fig.1 and Fig.2) and operating principles make it seem likely that if we study the analogue behaviour of the CMOS and TTL NOT gates in detail then we would expect to see differences, for example in terms of the input voltage at which the circuit can be definitely said to respond as a logic 1 input. As already indicated, CMOS device families are available which

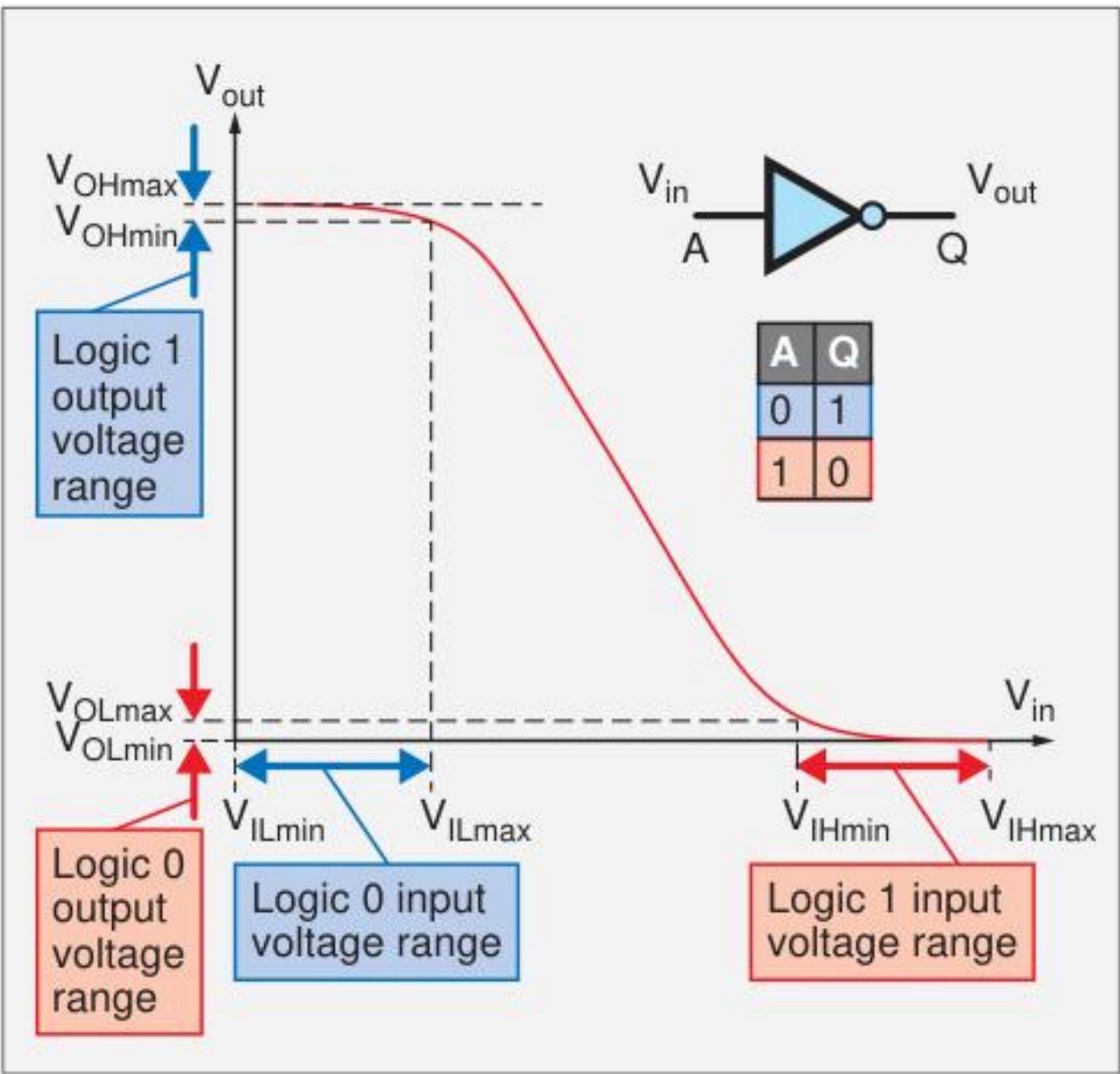


Fig.4. Transfer characteristic from Fig.3 showing ranges of voltages for the logical NOT operation.

operate at various supply voltages – so even chips using the same basic technology may have different logic voltages – particularly the list of voltages given earlier. On the other hand, TTL is well known for operating only on 5V supplies, but BiCMOS-based ‘low voltage TTL’ (LVTTTL) devices, such as the LVT family, operate on 3.3V.

We can consider a logic gate as if it is an analogue circuit and look at the input-to-output voltage relationships (transfer characteristic) – this is simplest for an inverter as it only has one input and will provide a general picture of the behaviour of other gates built with the same technology. A plot of an inverter transfer characteristic is shown in Fig.3 – this is just a sketch for illustration – it does not represent a real device or simulation data. The curve has a couple of horizontal regions where the output voltage does not change much and is either close to the supply voltage or ground. For logic operation we want the output in one of these parts of the curve. In between, there is a sloping region (in real gates it may be much steeper than shown). For logic operation we want to avoid this area, and also the transitions from the horizontal regions into the sloping section. The slope, at least the straight-line part, is like the characteristic of a linear amplifier and indeed it is possible to persuade some NOT gates to act as ‘analogue’ amplifiers (see last month’s *Magnetometer* project).

Fig.4 shows Fig.3 annotated with the ranges of voltages for which we could use this inverter as a NOT gate. Here, eight different voltages are labelled to define the limit of the four ranges shown. However, half of these limits are equal to supply or ground, so there are four key voltages that (along with the supply voltage) determine the switching operation of the gate:

- Maximum logic 0 input voltage (V_{ILmax})
- Maximum logic 0 output voltage (V_{OLmax})
- Minimum logic 1 input voltage (V_{IHmin})
- Minimum logic 1 output voltage (V_{OHmin})

In general, whatever the technology, logic gates should accept a given range of input

voltages as 1 or 0 and be guaranteed to produce a smaller range of possible output voltages, closer to the relevant ideal 1 or 0 voltage (supply or ground). This is seen to be the case with the transfer characteristic in Fig.4. This means that each gate tends to restore the voltage towards the ideal for that logic level. Although a

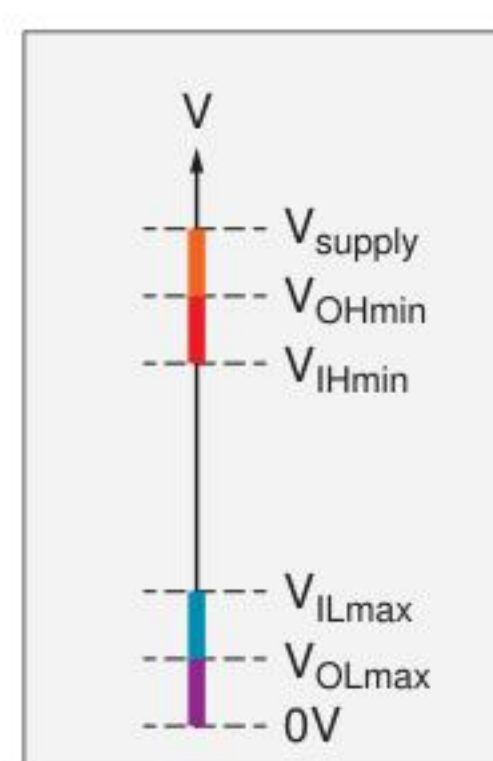


Fig.5 Graphical representation of logic gate voltages.

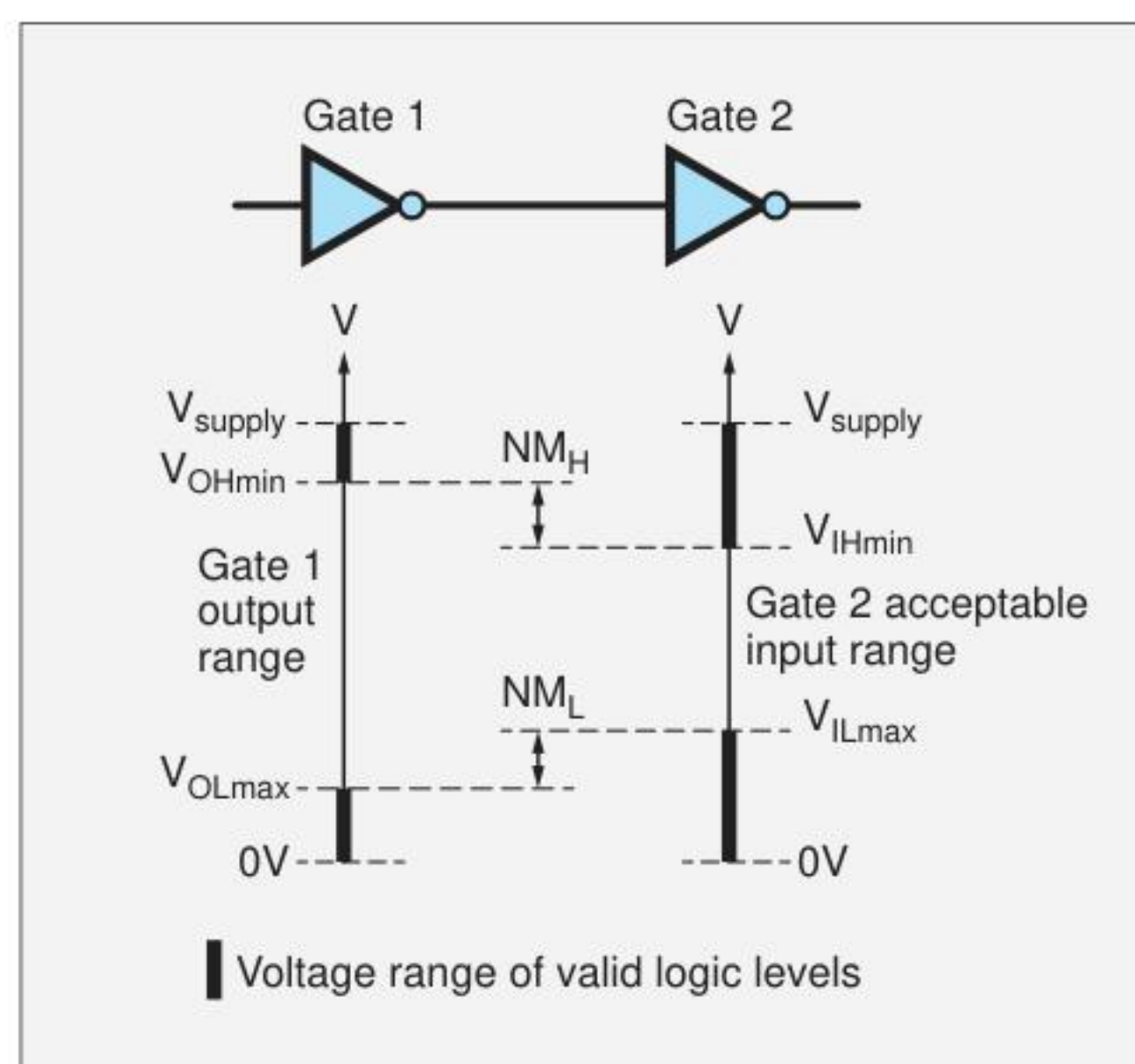


Fig.6. Logic levels and noise margins. The output range of a gate for logic 0 and 1 is smaller than the acceptable input range, thus restoring the voltage level to a more ideal one.

transfer curve is useful as a detailed representation of a gate’s characteristic, the full curve is not needed if we just want a graphical representation of the logic voltages – this is commonly done using a vertical bar or line on which the position of the relevant voltages are marked – see Fig.5.

The representation in Fig.5 also provides an immediate indication of the difference between the worst-case output level and worst acceptable input level for both logic 1 and logic 0. The numerical value for this difference is called the noise margin. More specifically, we can then define noise margins for logic 0 (NM_L) and logic 1 (NM_H) as follows:

$$NM_L = V_{ILmax} - V_{OLmax}$$

$$NM_H = V_{OHmin} - V_{IHmin}$$

This is illustrated in Fig.6. Noise margins indicate how well an interconnection in a digital circuit can cope with electrical disturbances (voltage shifts) without losing the correct value of the input data.

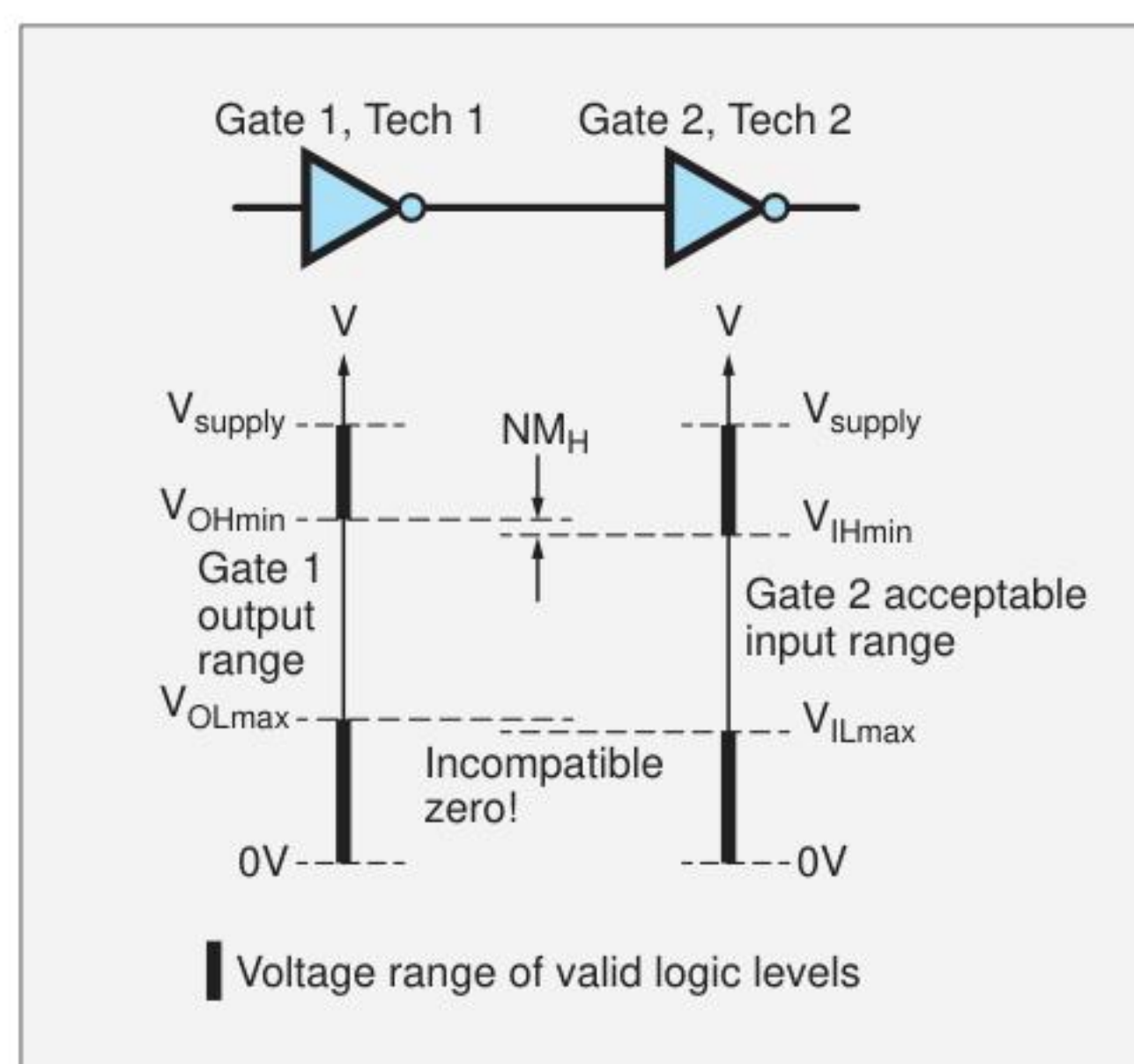


Fig.7. Example of incompatible circuits: different technologies on the same supply. Logic 0 output range of technology 1 is too large for technology 2 – not all valid 0 outputs from gate 1 will be recognised by gate 2. Logic 1 is OK, but the logic margin is very low, the logic 1 level will be sensitive to noise.

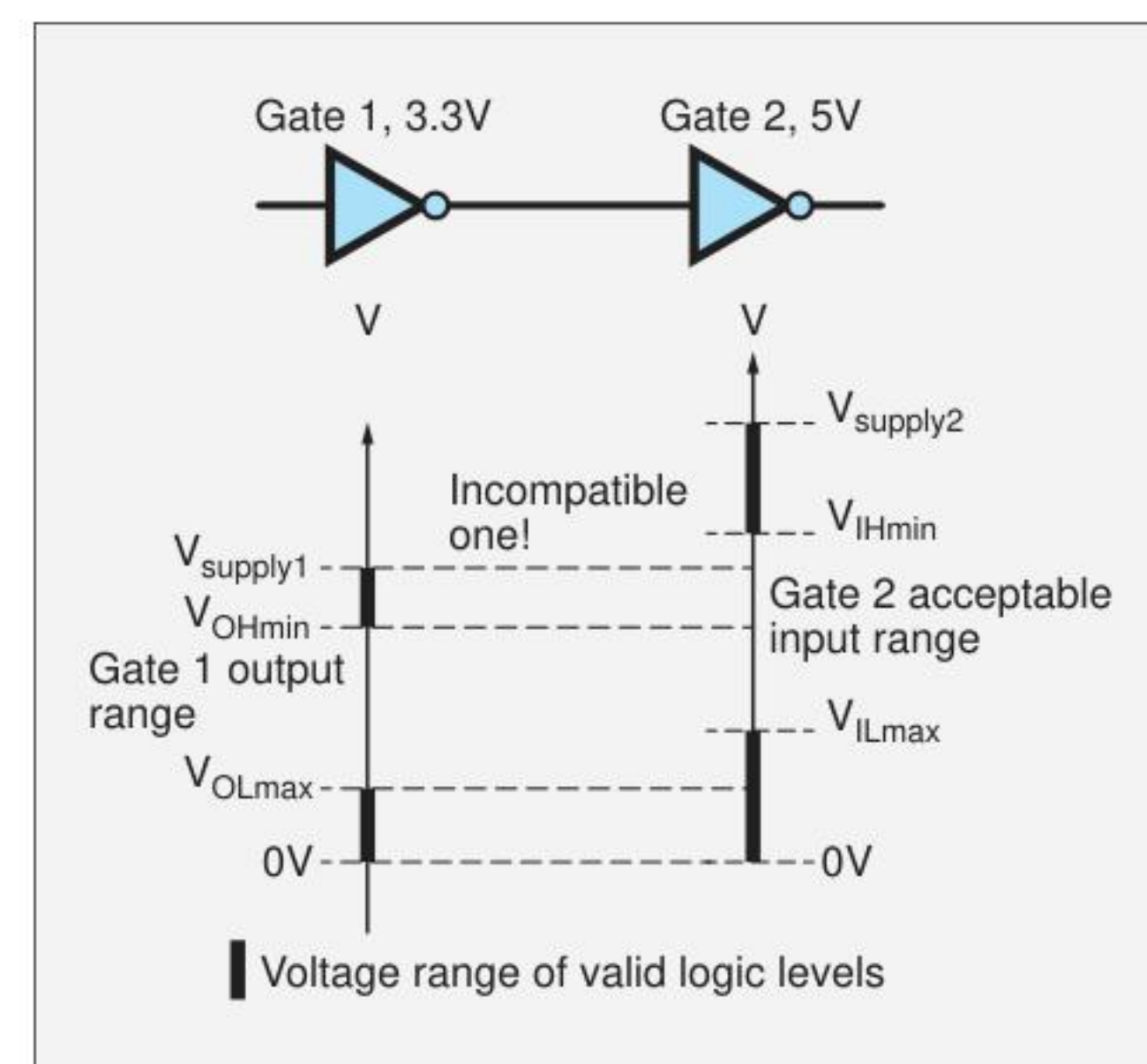


Fig.8. Example incompatible circuits: a low-voltage gate drives a higher-voltage gate. Logic 0 is OK, but the logic 1 output voltage is insufficient to be recognised by the second gate.

If the voltage is in between the defined logic levels for any reason (except for the brief time when switching between levels) then we have an undefined logic value and the circuit may behave unpredictably or even suffer damage.

Incompatible logic

Apart from noise issues, and as already indicated, there are two key things that can be different, and hence cause us potential interfacing problems when connecting digital circuits together. These are the logic technology (eg, CMOS and TTL) and the supply voltage. Different technologies may pose problems with incompatibility even on the same supply voltage. This is illustrated in Fig.7, which shows two possible problems in this situation – poor noise margin and mismatched logic ranges.

Fig.8 and Fig.9 illustrate possible problems when logic gates operating on different voltages are connected together. In Fig.8, a lower voltage gate

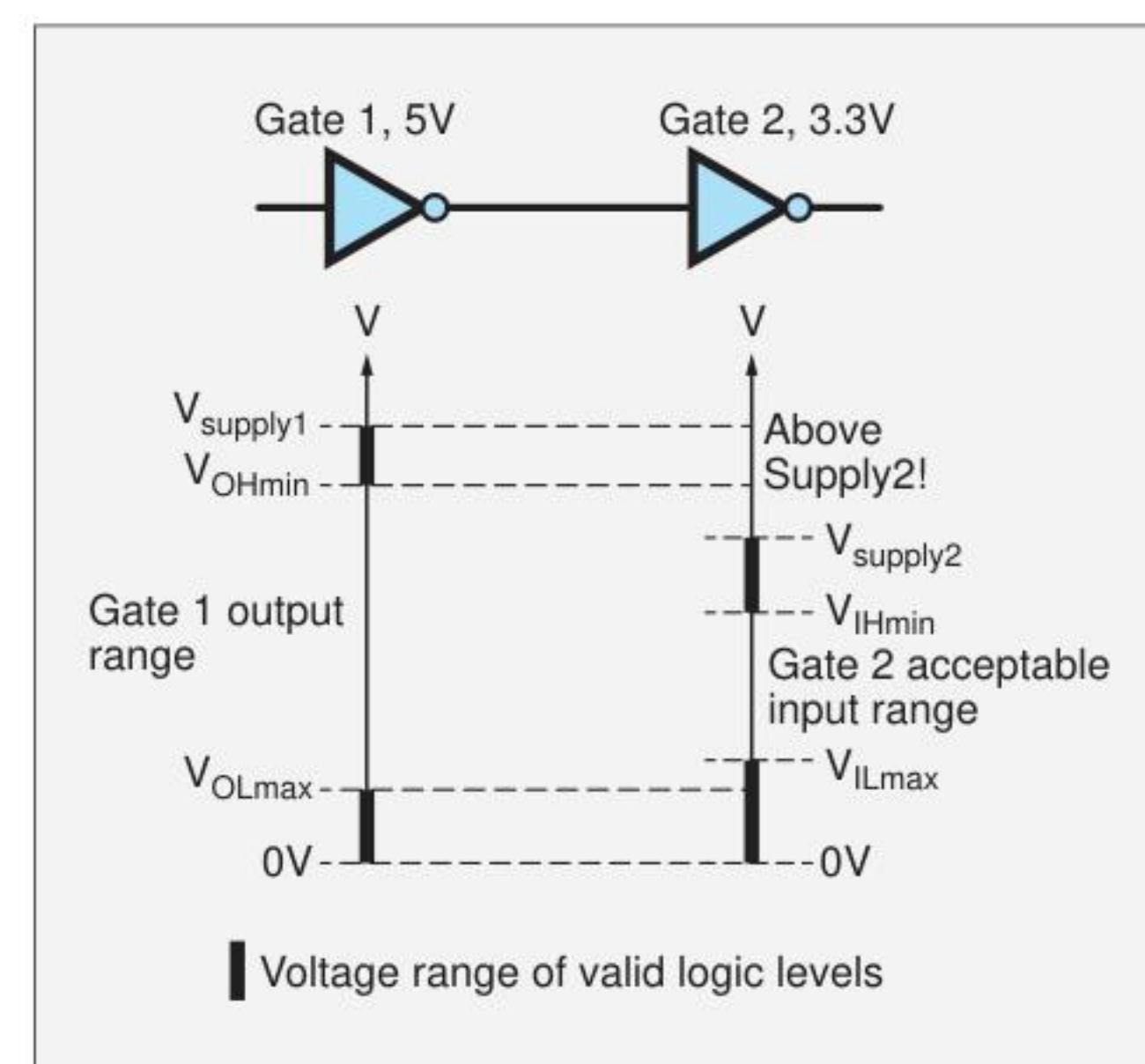


Fig.9. Example of incompatible circuits: a high-voltage gate drives a lower-voltage gate. Logic 0 is OK. The logic 1 output voltage is above the minimum logic 1 input requirement, but the voltage may be too high for the second gate.

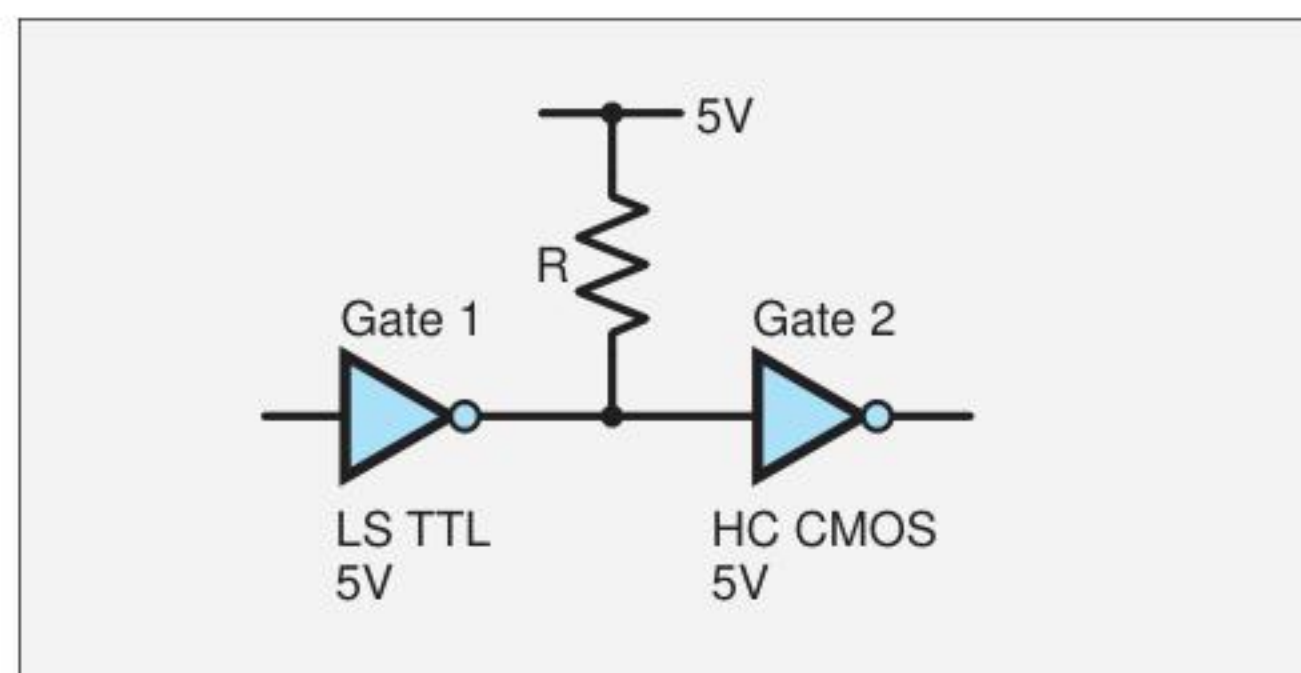


Fig.10. LSTTL-to-HC-CMOS interfacing.

drives a higher voltage one. The logic 0s are compatible, but the 1s are not. In this case, a logic 1 output from the first gate may hold the second gate in its intermediate voltage range, which as well as not being recognised as a valid 1 may cause other undesirable effects, such as causing higher-than-normal power dissipation in the second gate. In Fig.9, a higher voltage gate drives a lower voltage one. The logic 0s are compatible. For the logic 1, the output of the first gate is definitely above the minimum 1 for the second gate, so in that sense it is compatible; however, the high logic output voltage from gate 1 may damage gate 2. This situation is not necessarily always impossible – some digital inputs are specifically designed to cope with input voltages above supply voltage. This is known as voltage tolerance. For example, the situation in Fig.9 would be fine if the second gate was 5V tolerant. Information on input-voltage tolerance can be found on device datasheets.

When connecting gates of the same technology and supply voltage together we generally only have to worry about loading. The voltage ranges should automatically be compatible. However, for any technology there will be a limit to the number of inputs a gate's output can drive correctly. Loading effects will be relatively easy to evaluate for a single technology (with details readily available in datasheets), but when connecting two technologies together the situation may be more difficult to assess.

Connecting families – example

There are very many logic families that can potentially operate on the same voltage so we cannot discuss every possible interfacing situation. We will look at just one example – driving HC-CMOS from LSTTL. These are somewhat obsolete technologies, however, although they are still available. Furthermore, the discussion will serve as an example of some of the issues that occur with logic interfacing.

LSTTL specifications guarantee a 2.7V logic 1 output level, but HC-CMOS requires a 3.5V logic 1 input on a 5V supply. In practice, the LSTTL output will probably be sufficient, but compatibility is not guaranteed by the worst-case characteristics. To ensure compatibility, the output logic 1 from the LSTTL device can be raised by placing a pull-up resistor from the output to the supply voltage (see Fig.10). When the output goes high, the resistor pulls the voltage very close to the supply voltage. The value of the resistor should be chosen based on the number of other LSTTL inputs the gate is driving, in addition to CMOS input(s) (referred to as the LSTTL 'fanout') using the graph in Fig.11. For example, if an LSTTL device is driving only CMOS circuits then the LSTTL fanout is 0, so the resistor value is chosen from the left axis of the graph.

This data comes from a detailed application note on logic interfacing logic from Fairchild Semiconductor (now part of ON Semiconductor) (<http://bit.ly/pe-jan20-ttl>) which considers interfacing problems for a range of logic families available in the late 1990s and is a useful resource on this

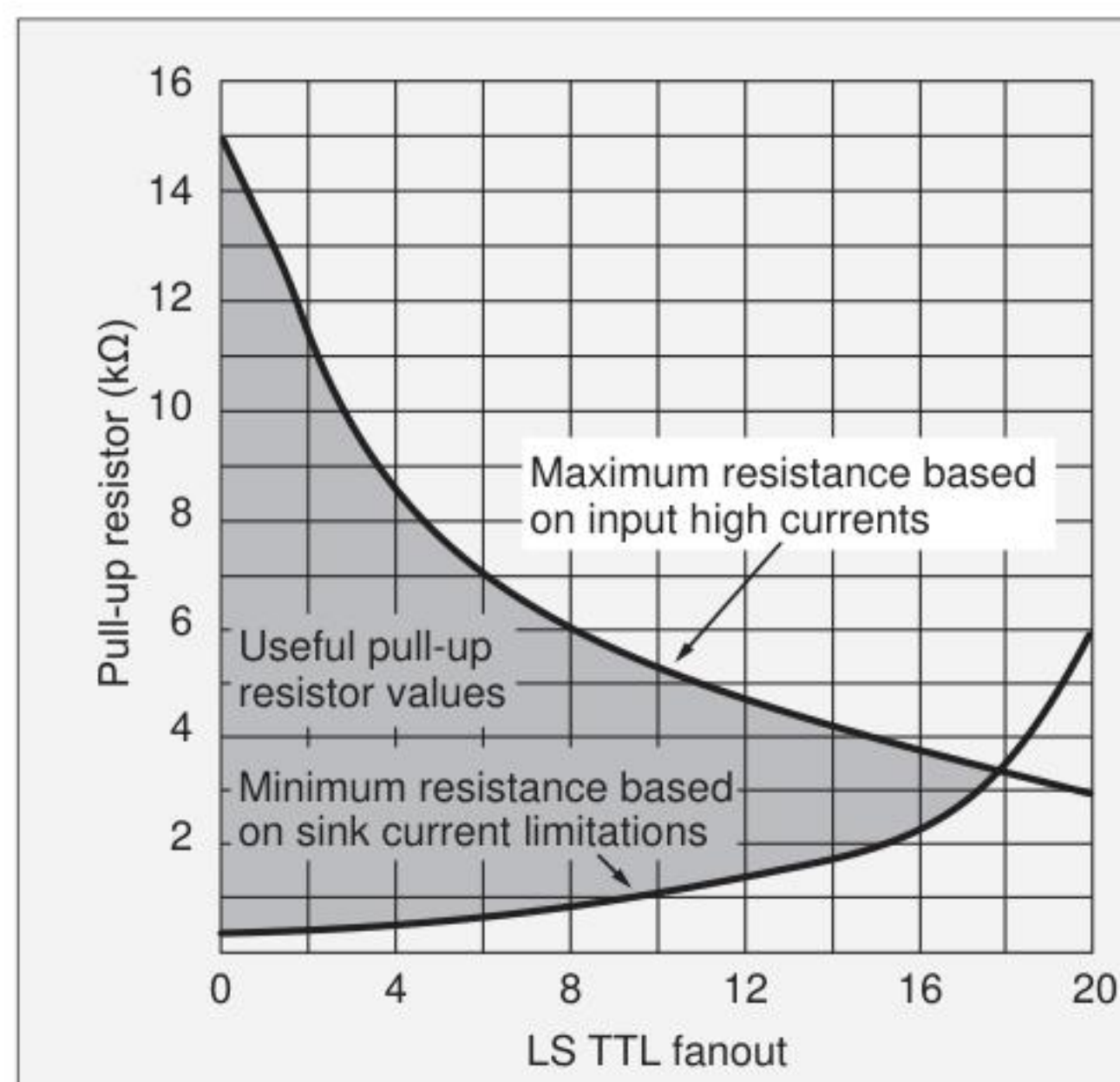


Fig.11. Choice of pull-up resistor for LSTTL-to-HC-CMOS interfacing (source: Fairchild Semiconductor application note AN-314 via ON Semiconductor).

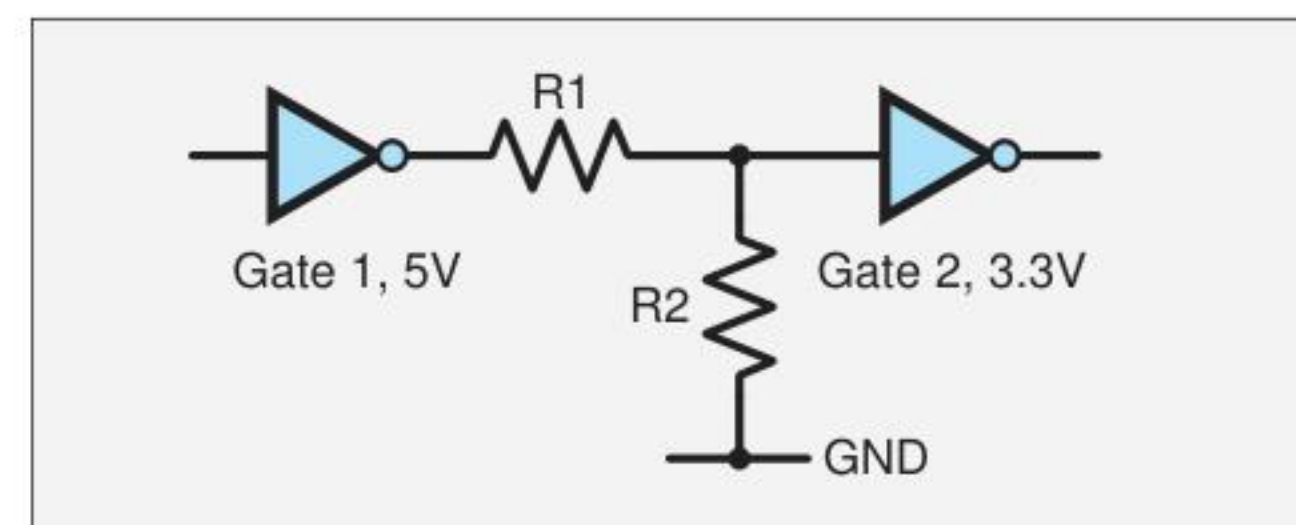


Fig.12. Potential divider logic interface.

topic for anyone interested in interfacing these older technologies.

When logic circuits are operating on different voltages, direct connection is not always possible and we need some kind of interface circuit. Perhaps the simplest is to use a potential divider to reduce the output level of a higher-voltage circuit

to make it compatible with a lower one (see Fig.12). Typical values might be $R1=18\Omega$ and $R2=33k\Omega$ for 5V to 3.3V. Use of a potential divider will increase power consumption and may result in slower switching.

Level translation ICs

In this article we have covered the basics of logic voltage levels and some of the problems that may occur when interfacing logic circuits using different technologies and supplies. This is a common problem in digital design and there are therefore various ICs available which are specifically aimed at logic-voltage translation – these include translation of bidirectional signals as well as the single direction connections we have been discussing. Using these ICs is often the best approach.


CRICKLEWOOD ELECTRONICS
Established 1981

Your best bet since MAPLIN
Chock-a-Block with Stock

Visit: www.cricklewoodelectronics.com

Or phone our friendly knowledgeable staff on **020 8452 0161**

Components • Audio • Video • Connectors • Cables
Arduino • Test Equipment etc, etc



Visit our Shop, Call or Buy online at:
www.cricklewoodelectronics.com

020 8452 0161

Visit our shop at:
40-42 Cricklewood Broadway
London NW2 3ET

Make it with Micromite

Phil Boyce – hands on with the mighty PIC-powered, BASIC microcontroller

Part 12: Writing serial data to external modules

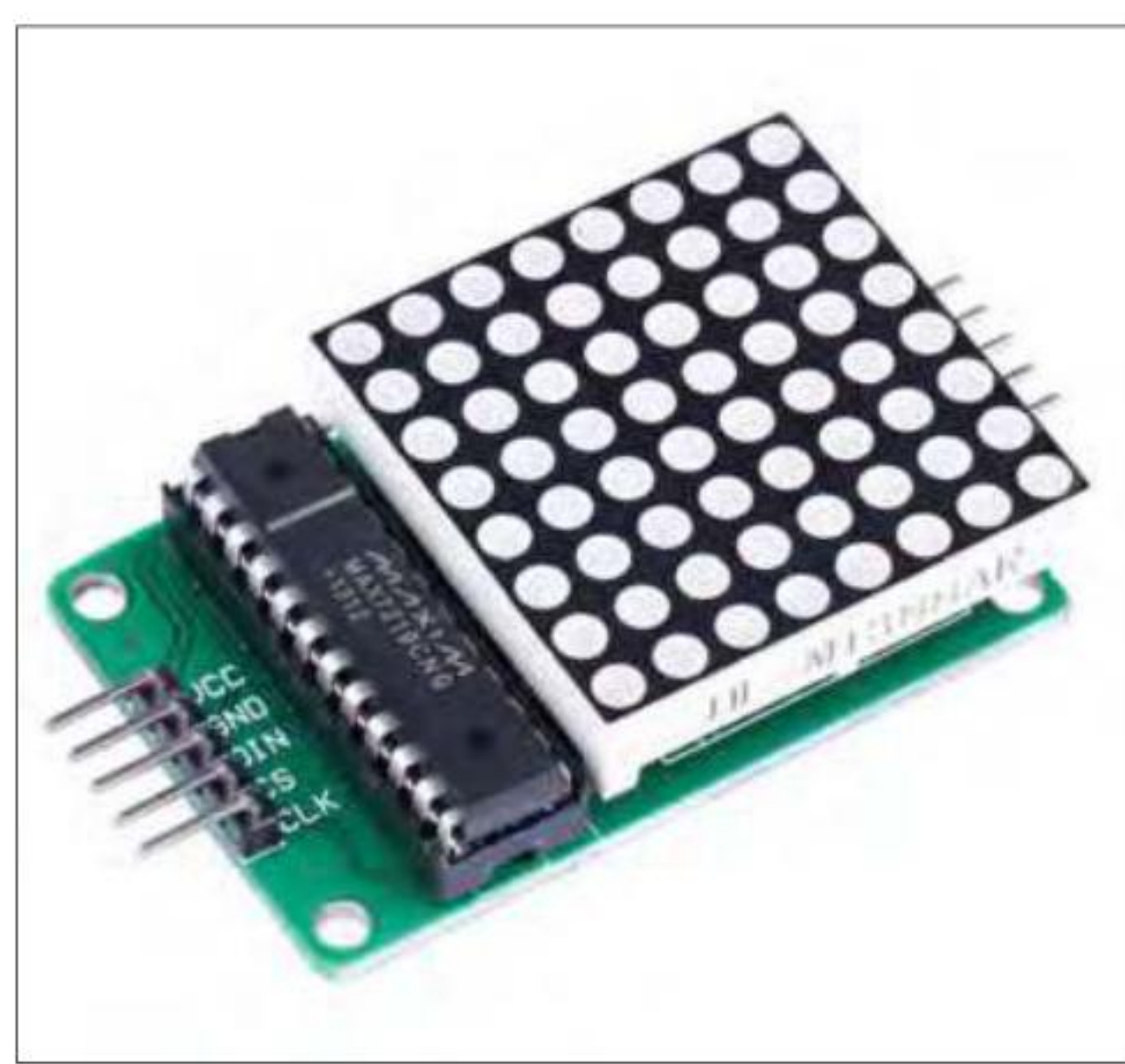


Fig.1. This 8×8 LED matrix module is based on a MAX7219 IC. It is ideal for demonstrating how to use the Micromite to write serial data to external hardware.

Last month, in Part 11, we discussed the concepts behind serial data communication. We demonstrated how to *read* the serial data generated from a 4×4 capacitive keypad module in order to determine which numbered pad had been pressed. This month, we will continue to discuss serial data communication; however, the focus will be on *writing* serial data to an external module in order to control it. To help demonstrate this, we will use an 8×8 LED matrix (see Fig.1). We will write a program to alter the logic levels bit-by-bit on the serial connection – a method known as ‘bit-banging’.

MAX7219 LED matrix module

An 8×8 LED matrix contains a total of 64 LEDs. To control all these LEDs individually would require 64 I/O lines; something our 28-pin Micromite does not have. Even if we were to use a 144-pin Micromite eXtreme (yes, it does exist), the

result would be a wiring nightmare. An alternative method is to drive one single column of eight LEDs (which requires only eight I/O lines) and then repeat this quickly for the other seven columns by enabling them one at a time (requiring a further eight I/O lines) – this is called ‘multiplexing’. The key benefit of multiplexing is that it reduces the number of I/O lines needed from 64 to just 16; and this is something that we could manage with the 28-pin Micromite (which has 19 available I/O lines). However, this article is all about writing serial data and so we have chosen this particular 8×8 LED matrix module because it contains a MAX7219 driver chip which uses a serial input connection to control all 64 LEDs. The MAX7219 contains an 8×8 static RAM that stores each LED’s status (on or off), and it also contains the required multiplexing circuit to minimise wiring to the LEDs themselves; yet it can all be controlled as simply as if

it were a serial shift register. This means we can control *all* 64 LEDs with just three pins for the serial link (plus power) from the Micromite.

Fig.2 shows a simplified block diagram of the MAX7219 LED matrix module. The two main components are IC MAX7219 and the 8×8 LED matrix display. These are connected together via 16 PCB tracks (eight rows, eight columns). There are just five external connections required – two for power (3.3V or 5V can be used, plus ground), and three for serial communication. The serial link comprises the Micromite (acting as the transmitter), and the MAX7219 (acting as the receiver). It is a *synchronous* link, meaning that there is a data signal (DIN – data in) and a clock-signal (CLK) – refer to last month for more details. In addition, there is a Chip Select ($\overline{\text{CS}}$) signal. Note the bar over $\overline{\text{CS}}$, it is *active low*. The relative timing of logic levels on these three signals is shown in Fig.3.

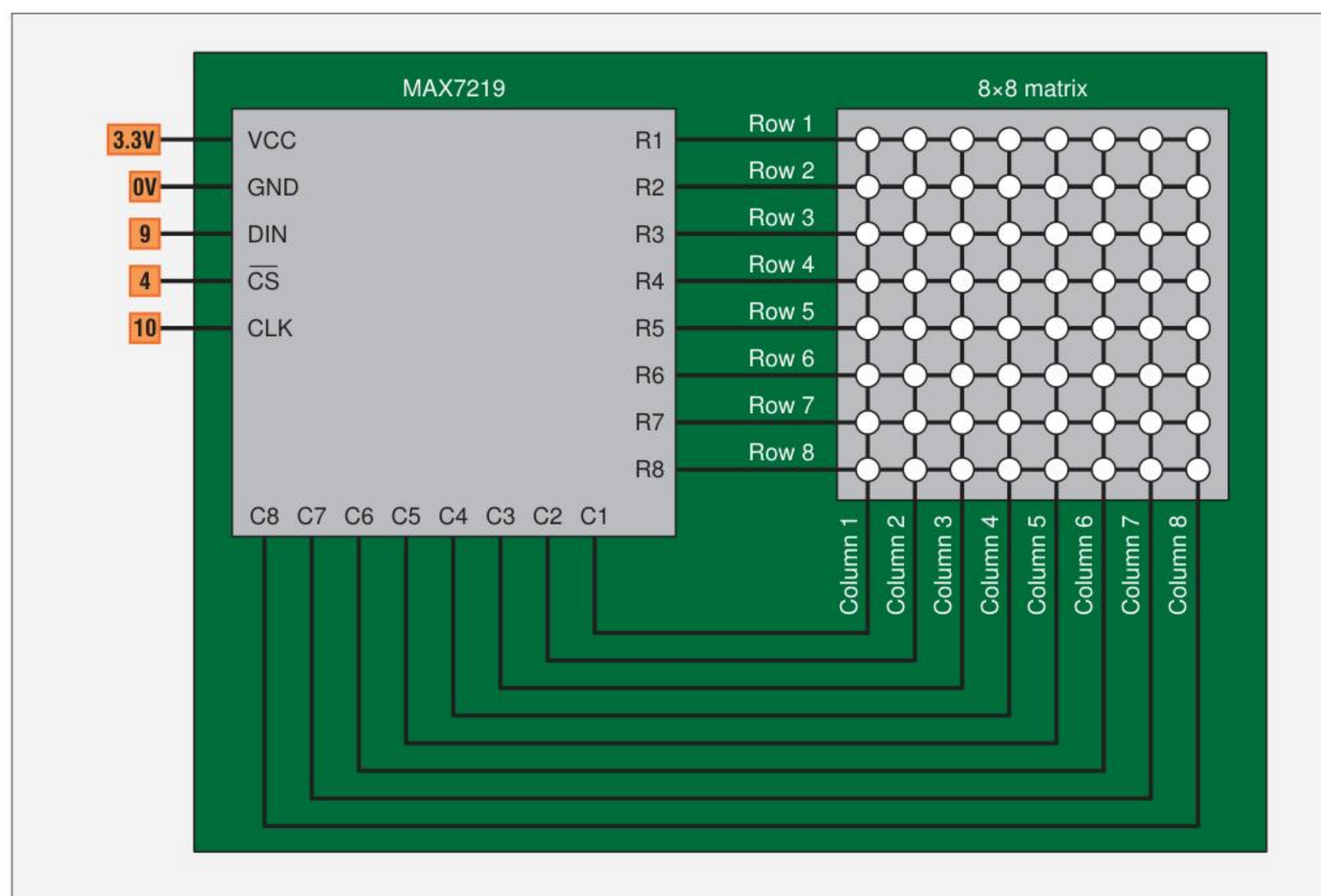


Fig.2. Simplified block diagram of the MAX7219 8×8 LED matrix module. Just five external connections are required to control all 64 LEDs; shown here are the connections to the Micromite to bit-bang data to it.

Micromite code

The code in this article is available for download from the *PE* website.

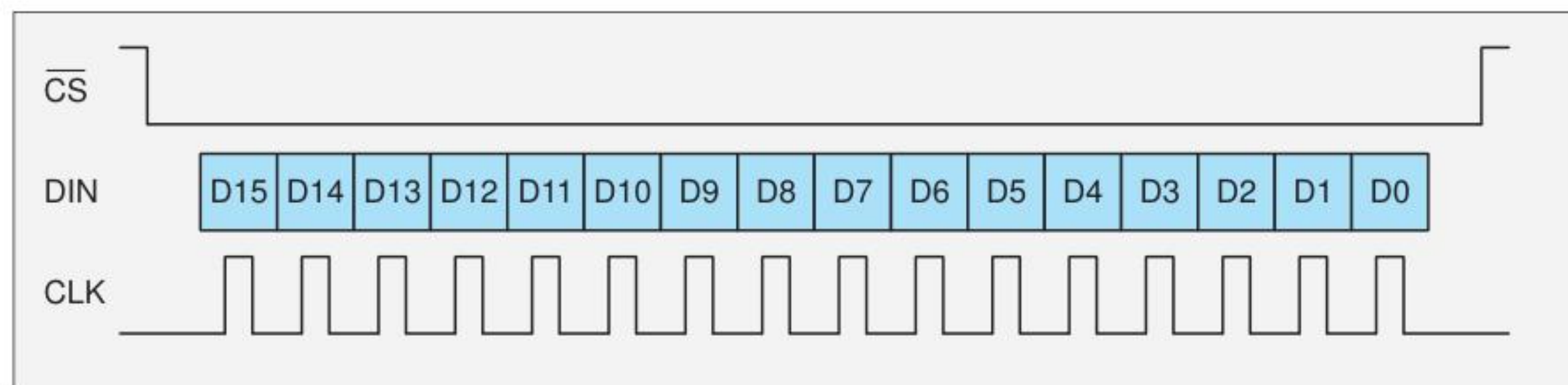


Fig.3. The serial link to the MAX7219 can be regarded as a 16-bit shift register. The 16 data bits are clocked into the MAX7219 while the Chip Select ($\overline{\text{CS}}$) pin is held low.

Controlling the MAX7219

In essence, the MAX7219 can be regarded as a 16-bit shift register. For the Micromite to send 16-bits to the MAX7219, the following steps need to occur in the following sequence (also refer to Fig.3):

1. Set $\overline{\text{CS}}$ signal Low
2. Set DIN signal high or low, as required
3. Pulse the CLK signal high
4. Repeat steps 2 and 3, 16 times
5. Set $\overline{\text{CS}}$ signal high

The above can be written in code by using the commands `SETPIN(x), DOUT, PIN(x)=`, and `PULSE x`, (and embedding most of these in a `FOR/NEXT` loop). However, before we write any code, we first need to understand the structure of the 16 bits. From the MAX7219 datasheet, we can see the 16 bits include a 4-bit register-address (green), an 8-bit data value (blue) and four bits which are unused (red). This is summarised in Fig.4.

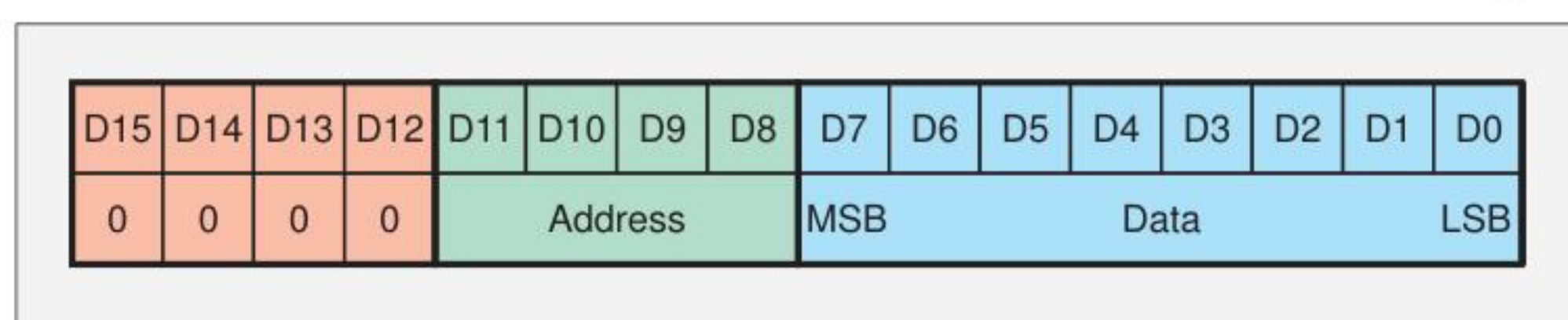


Fig.4. The structure of the 16 bits sent to the MAX7219. It comprises four (unused) leading 0s, then a 4-bit register address, and finally an 8-bit data value.

We now know the structure of the 16 bits, next we need to understand what the different registers do, and the valid data values each register can take. This is summarised in Fig.5.

Referring to Fig.5, you will see that the first eight registers control the LEDs in the eight LED columns. These eight registers are the ones we will need to use in order to control the LEDs. Five other registers are also used, and these either need to be set to a specific value (as shown), or can be set to any valid value. Take for example the BRIGHTNESS register; it can take any value between 0 and 15 (so setting the BRIGHTNESS register to 15 will set the overall LED brightness to a maximum). These five registers can be considered as 'one-time set up' registers and only need to be set once at the beginning of our program, unless you wish, for example, to change the brightness during use.

We are almost in a position to start writing some code; however, let's first understand how the 16 bits can be represented by a hexadecimal number.

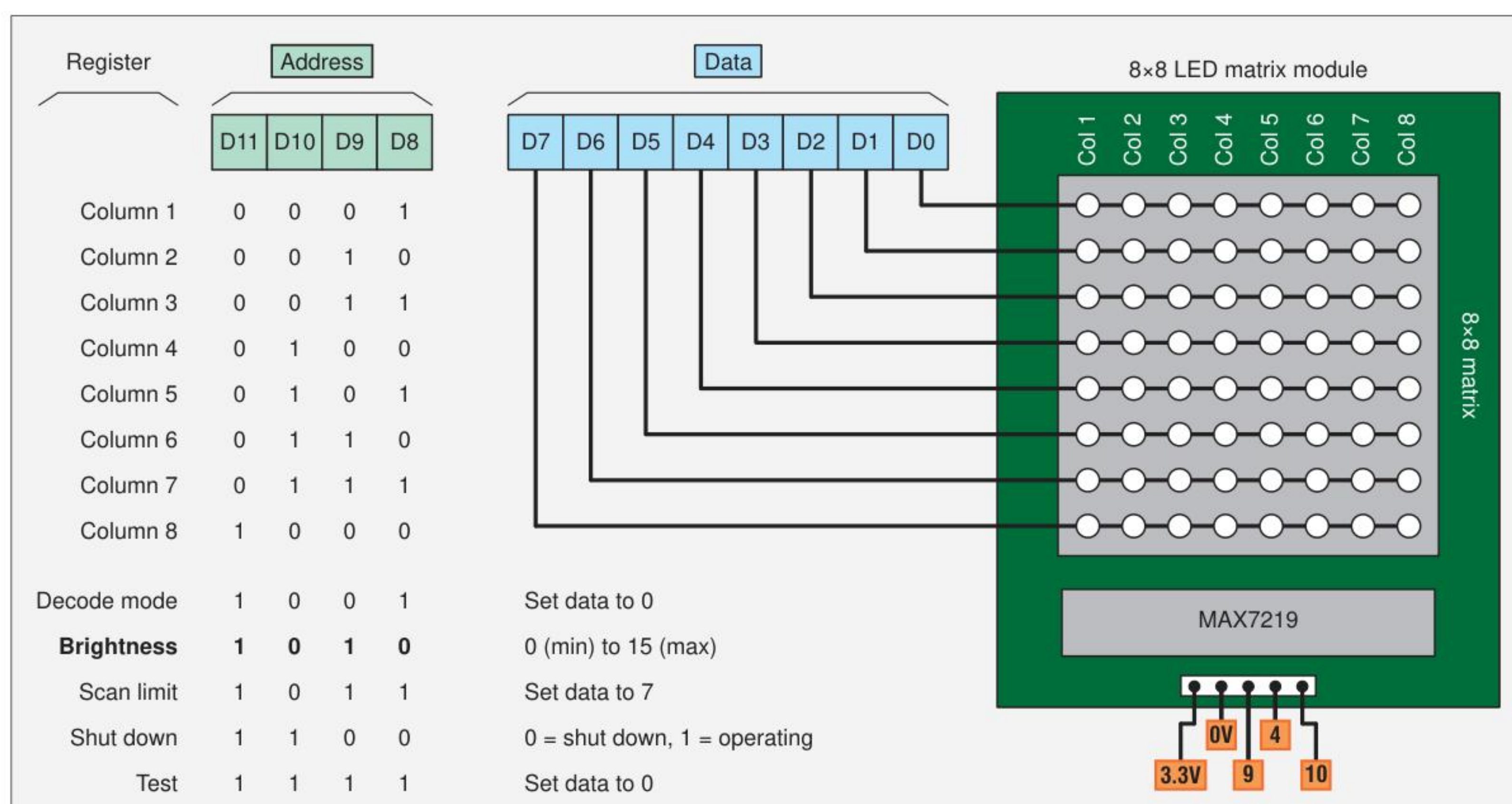


Fig.5. The register addresses used to control the 8x8 LED matrix; and the data-values that can be sent. Registers 1 to 8 are connected to LED columns 1 to 8. To illuminate an LED in a column, the relevant data-value bit (D0-D7) needs to be set to 1.

Using hexadecimal

This is not going to be an exhaustive tutorial, we just want to highlight how to use hexadecimal (hex) numbers to simplify (and represent) the 16 bits we send to the MAX7219.

In *decimal*, there are the ten unique digits that we are all familiar with: 0-9. In *binary*, there are just two unique digits: 0 and 1. However, in *hexadecimal* there are sixteen unique digits; 0-9, and also A-F (representing decimal values 10 to 15). Any hex digit can also be represented as a 4-bit binary number ranging from 0000 to 1111. The bottom part of Fig.6 uses illuminated LEDs to represent a binary 1, and if you refer to it you can see how any 4-bit binary number relates to a single hex digit. Take for example hex digit B, this equates to the 4-bit binary value 1011.

The reason for explaining this is that the 16 bits we send to the MAX7219 can be considered as four 4-bit binary values. In turn, this equates to four hex digits; an example will help to explain this. Let's say we want to set the BRIGHTNESS to a value of 2. Referring to Fig.4 and Fig.5 we can see that we need to send the following 16-bits:

	ADDRESS	DATA VALUE
D15-D12	D11-D8	D7-D0
0 0 0 0	1 0 1 0	0 0 0 0 0 0 1 0

Using Fig.6, we can see that it is equivalent to the four hex digits: 0 A 0 2. In MMBASIC, hex numbers are prefixed with `&h`, so the required number is written as `&h0A02`.

As a further example, we will now look at the eight hex numbers required to turn all four corner LEDs on (and switch all other LEDs off). Work through each 16-bit value yourself to ensure that you understand what is happening (some useful comments are included):

- `&h0181` register 1 (= column1), data value = `&h81` = 1000 0001 (D7=bottom LED, D0=top LED)
- `&h0200` register 2, all data 0, so all LEDs off
- `&h0300`
- `&h0400`
- `&h0500`
- `&h0600`
- `&h0700`
- `&h0881` register 8 = column 8. Top / bottom LEDs

The above should provide enough information to understand what 16-bit data values to use in order to control all aspects of the LED matrix module.

Extracting the data bits

We will now discuss how to extract each of the 16 data bits (in sequence D15 to D0) from the hex number we are sending to the MAX7219. To do this we will be using the logical AND command. Referring back to Fig.3, each bit is used to set the DIN pin to the correct logic level prior to pulsing the CLK pin. In the following discussion we will continue to use the previous BRIGHTNESS=2 value: &h0A02. We will also be using the x^y operator, which raises x to the power of y . In terms of binary this means 2^y (stick with me – I promise all will become clear). Below, the 16-bit binary representation of 2^y is shown for all y values from 15 down to 0.

As you can see, it sets just a single bit high with all other 15 bits remaining low.

D15-D12	D11-D8	D7-D4	D3-D0
0000	1010	0000	0010
$2^{15} = 1000$	0000	0000	0000
$2^{14} = 0100$	0000	0000	0000
$2^{13} = 0010$	0000	0000	0000
$2^{12} = 0001$	0000	0000	0000
$2^{11} = 0000$	1000	0000	0000
$2^{10} = 0000$	0100	0000	0000
$2^9 = 0000$	0010	0000	0000
$2^8 = 0000$	0001	0000	0000
$2^7 = 0000$	0000	1000	0000
$2^6 = 0000$	0000	0100	0000
$2^5 = 0000$	0000	0010	0000
$2^4 = 0000$	0000	0001	0000
$2^3 = 0000$	0000	0000	1000
$2^2 = 0000$	0000	0000	0100
$2^1 = 0000$	0000	0000	0010
$2^0 = 0000$	0000	0000	0001

Bit-banging the MAX7219

Now consider the following six lines of code (while referring to Fig.3, and also the above table). This will send the 16-bit value (&h0A02) to set the BRIGHTNESS value to 2.

```
PIN(ChipSelect) = 0
FOR mask = 15 TO 0 STEP -1
    PIN(SerialDataOut) = (&h0A02) AND (2^mask)
    PULSE Serial_Clk, 0.05
NEXT mask
PIN(ChipSelect) = 1
```

The first line sets the ChipSelect signal low so that the MAX7219 responds to clock pulses on the module's CLK pin. The second line defines a FOR/NEXT loop, which repeats 16 times – it uses a variable mask, set initially to 15 (and works down to 0). The third line is the clever line of code. It takes the hex number &h0A02 and applies a logical AND with 2^{mask} . So on the first time through the FOR/NEXT loop, (&h0A02) AND (2^{15}) will extract D15 from the hex number &h0A02 (which is 0) and this value is passed onto the SerialDataOut pin. The fourth line of code pulses the Clk signal high, which loads D15 (first time through FOR/NEXT loop) into the MAX7219. All 16-bits

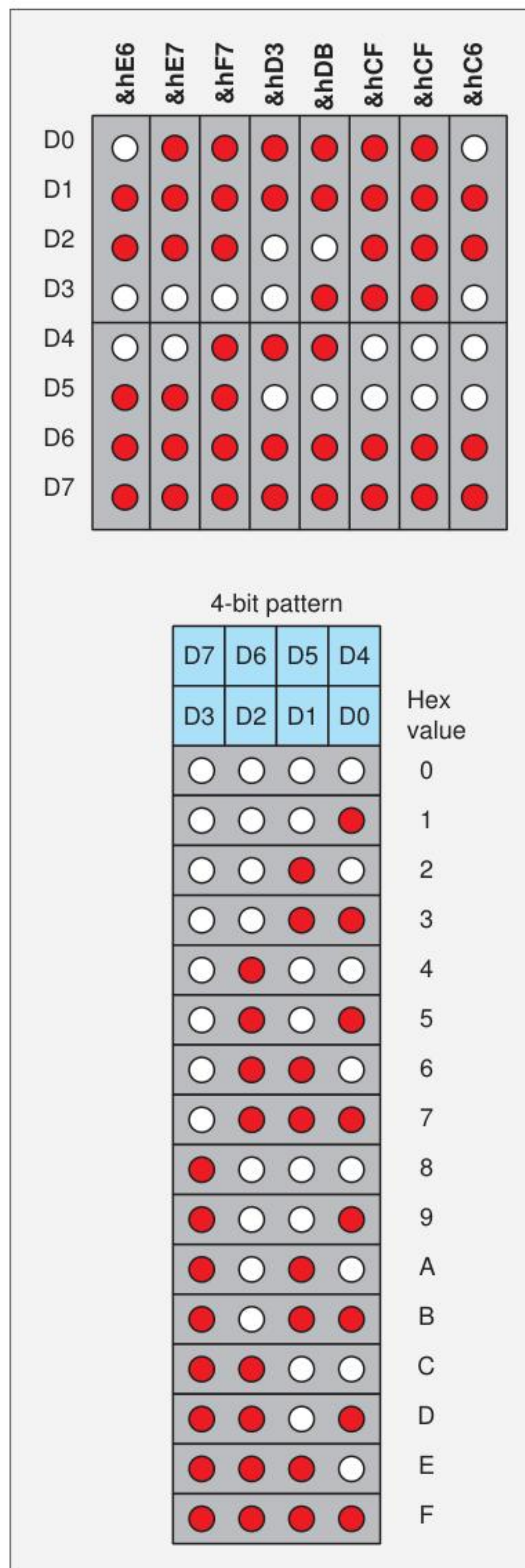


Fig.6. The eight hex numbers that represent the character '2'. Use the look-up table to see how other hexadecimal values can generate any pattern of dots.

of &h0A02 are extracted in this method. So, on the fifth excursion through the FOR/NEXT loop, mask will have a value of 11, and (&h0A02) AND (2^{11}) will extract D11 from &h0A02, which is 1. This sets the SerialDataOut pin high, and gets clocked into the MAX7219 with the PULSE line of code. When all 16 bits have been clocked into the MAX7219, the last line of code returns the ChipSelect pin back high, which completes the serial data transfer.

To make the above code more flexible, we can put these six lines of code into a subroutine, to which we can pass our hex number that needs to be bit-banged to the MAX7219. You will see this shortly as it is the method used in the downloadable program code.

Displaying numbers

Now that we know how (and what) to communicate with the LED matrix module, we can create a useful program. We will begin with code to repeatedly display the digits 0 through to 9 in turn – not very useful, but it will help demonstrate a few useful things! The program to do this (MIWM_LEDmatrixCount.txt) can be downloaded from the January 2020 page of the PE website. The program is commented throughout, and along with what has been discussed so far, you should be able to understand most of the code.

Before running the program, connect a MAX7219 8x8 LED matrix module to the Micromite/Development Module. The DM plugs into a breadboard (via downward facing pins next to J12) along with the LED matrix module (see Fig.7). Then it is just a matter of using five jumper wires to make the required connections, as shown in Fig.2. With the software installed (AUTOSAVE, paste, Ctrl-Z) and the LED matrix module connected, RUN the program. If all has gone well, you will see the digits 0-9 counting up on the LED matrix. If you don't see this, then there is not much to check – simply ensure you have the five connections made to the correct five pins, and that the correct code is loaded (go into the Editor, and check that the first few lines of code read: Make-it-with-Micromite (Part 12) Bit-Banging MAX7219 8x8 LED Matrix.

Once you're up and running, there are some points worth mentioning about the code. Stop the program (Ctrl-C) and take a quick look at the code. Working down, you will see the following:

- The SETUP section begins by setting the three pins used for the serial link
- A two-dimensional array, Pattern(10,8) is declared – it stores the eight hex numbers for each of the ten digits, 0-9 (more on this below)
- The Pattern array is filled with the required dot patterns for the ten digits using DATA and READ
- Subroutine M7219_INIT is called, which initialises the MAX7219 (sends the set-up bytes, including setting BRIGHTNESS to 2)
- The MAIN PROGRAM continually loops from 0 to 9, setting variable DispCount. It passes the value in DispCount to the subroutine LoadPattern (explained below)
- The SUBROUTINES section contains four subs. The last one is called MAXwrite and this contains the six lines of bit-banging code discussed above. The only difference is that the hex number passed to it is stored in a variable named value (rather than it being the fixed number &h0A02 used above)
- Subroutine M7219_INIT sends the five initialisation values shown in Fig.5. It then calls the subroutine ERASE_ALL
- Subroutine ERASE_ALL simply sends &h00 to all eight LED columns (ie, to registers 1-8). The equation column*256 ensures the correct bits are set on D8-D11 (see Fig.5). Tip: If you

are not familiar with binary, then use the Calculator built into Microsoft Windows or macOS. Set it to 'Programmer' mode (as opposed to 'Standard' in Windows, 'Basic' in macOS) and then you can switch between decimal and binary (and even hex). As an example (and using 'column' with a value of 2) – in decimal mode, type $2 \times 256 =$ and then observe the binary equivalent that is shown (0010 0000 0000). This shows you that D9 is set in our 16-bit number, which represents Column 2. The data bits are all 0, resulting in all LEDs off in Column 2 (once this number has been bit-banged to the MAX7219 by calling subroutine MAXwrite)

■ The subroutine LoadPattern loads variable ww with the digit value we wish to display. A FOR/NEXT loop is then worked through eight times using variable i (set initially to 1, working up to 8). Within the FOR/NEXT loop, the two-dimensional PATTERN array is used to lookup the eight hex numbers representing the dot pattern for digit ww. These are then passed, one at a time to the relevant Column register using the formula: (256*i)+Pattern(ww,i)

The above has hopefully made it clear how the program works. The most difficult part to understand initially is how the dot patterns are recalled and passed to the MAX7219. We will now explain a little more as to how the dot patterns are implemented so that you can then modify them in the program to display your own characters. In the following discussion we will specifically look at the digit '2'.

Referring to Fig.6 (top part), you can see that the eight dot pattern values for each of the eight columns are: &hE6, &hE7, &hF7, &hD3, &hDB, &hCF, &hCF and &hC6. Now take a quick look at the SETUP section of the code. There are ten DATA lines containing the dot patterns for the ten digits. The third DATA line represents the digit '2' and you will see that it contains these exact same eight dot-pattern numbers. They are loaded into a two-dimensional array in the SETUP section of the code (by reading all the DATA values into the array Pattern). The eight dot-pattern numbers discussed here for the digit '2' are stored in Pattern(2,1), Pattern(2,2), Pattern(2,3) ... all the way to Pattern(2,8). Using an array like this makes the dot-pattern values easy to recall with a FOR/NEXT loop (as is done in SUB LoadPattern).

Let's now look at how the dot-pattern values are created – it is quite easy, especially if you are familiar with binary. Sticking with digit '2', we will examine the dot pattern for the first column – the value &hE6. Fig.6 shows each of the eight columns as two blocks of four LEDs – these

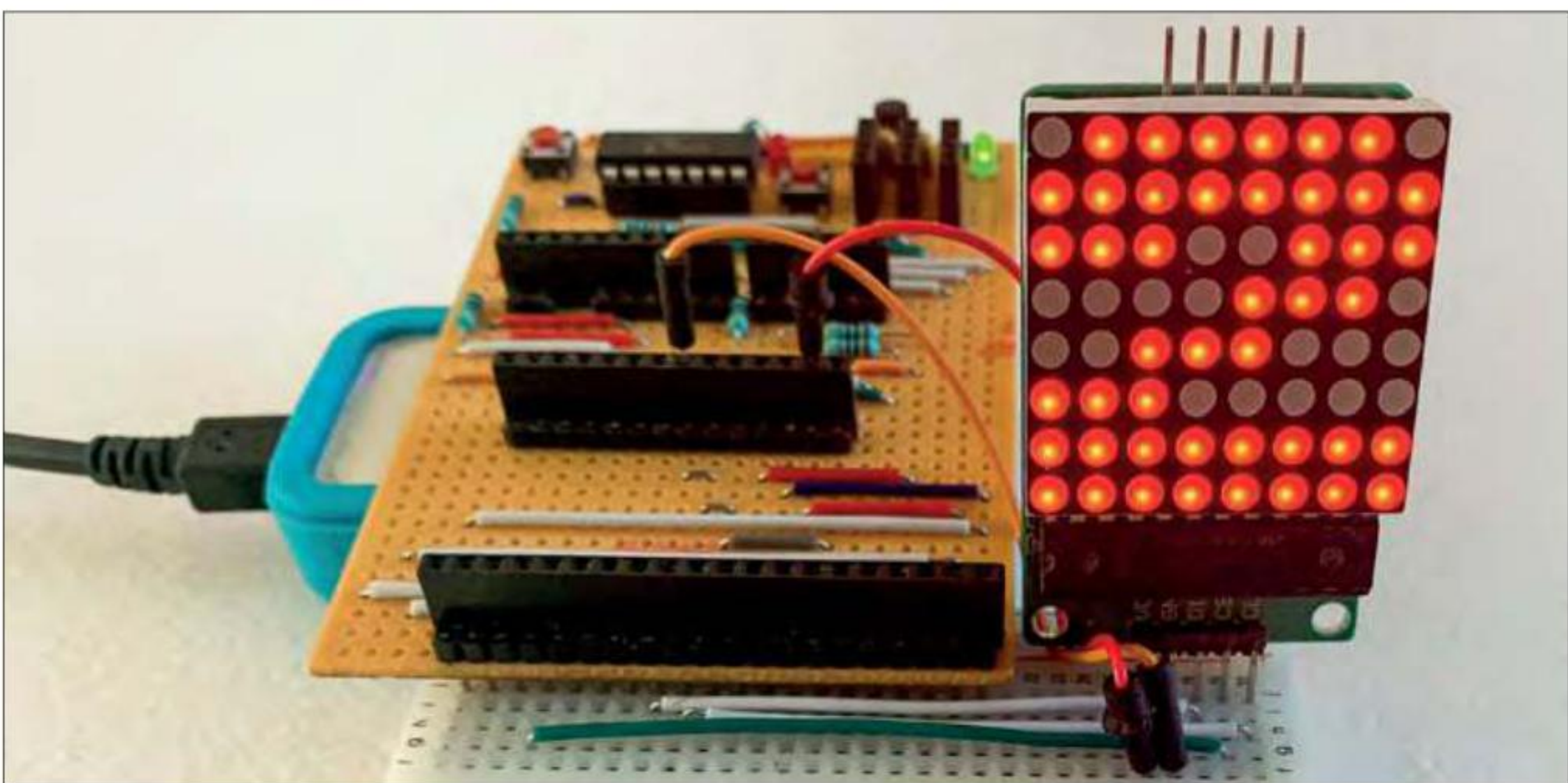


Fig.7. Breadboard and five jumper wires connect the 8x8 LED module to the Micromite.

are [D7, D6, D5, D4] and [D3, D2, D1, D0] as labelled on the left of the diagram. Take the first (vertical) LED column and write it out horizontally as follows:

D7	D6	D5	D4	D3	D2	D1	D0
1	1	1	0	0	1	1	0

Now take these two 4-bit blocks and use the bottom part of Fig.6 to convert them into hex values. You will see that the first block [1 1 1 0] converts to hex value E, and the second [0 1 1 0] converts to hex value 6; hence the dot-pattern value is &hE6. Passing this to the relevant Column register (1) will set the LEDs as required.

Let's try this now. RUN the program, then stop it with Ctrl-C. Then at the command prompt, type ERASE_ALL to clear the matrix. If the matrix doesn't clear, you probably launched the Editor, which will prevent this from working. If this is the case, RUN / Ctrl-C again to return to the command prompt. Now type: MAXwrite(&h01E6) and you should see the dot pattern &h01E6 for digit '2' appear in the first column (&h01E6).

You should now see that to send dot pattern XY to column C, simply type MAXwrite(&h0CXY) (replacing C with a value between 1 and 8, and X and Y with the relevant hex digits) – try it! If you're still unsure, type MAXwrite(&h08FF) to turn on all the LEDs in the last column.

Viewing in slow motion

To see the eight dot-pattern values being sent one at a time to build up the character displayed on the matrix, uncomment the two lines indicated in the Main Program (simply remove the ` at the start of both lines, and then press F2 to RUN). d=500 adds a 500ms delay between each column of LEDs being set, and ERASE_ALL clears the matrix before the next character is drawn. Re-comment both lines once you're done.

Things to try

A lot of detail has been covered and by now you should be comfortable with controlling

the MAX7219 8x8 LED matrix module. To test your knowledge, here are a few things to try (starting easy, and getting harder)

1. Speed up the count (Hint: change the delay in the Main Program)
2. Count down as opposed to counting up (Hint: FOR DispCount STEP -1)
3. Instead of counting numbers, display the letters A-Z instead
4. Make 10 smaller digits occupying three dots wide and five dots high (Hint: change all dot-pattern values in DATA)
5. Count from 00 to 99 using the smaller, digits (Hint: SUB LoadPattern needs to load two digits, each being three dots wide with a blank column between them to separate the digits)
6. Make an animation (Hint: create the dot patterns for a few custom characters with each character changing only slightly. Then sequence them with a small delay to create an animated effect)

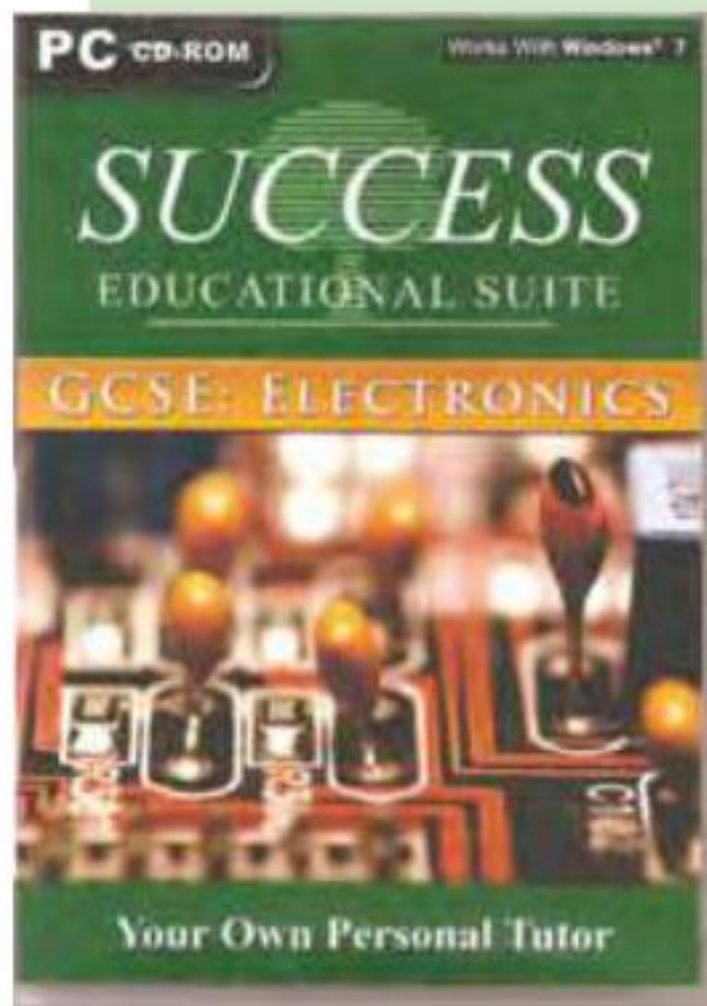
Idea number 6 above can be really good fun to play with and is actually quite straightforward to achieve. As an idea to explore, try drawing a face (dots for eyes, nose, and mouth) and create several more faces each one with something slightly different – blink the eye(s), move the mouth (happy, sad, neutral). Then play these in a random sequence, and with random delays between them. This can result in a fairly realistic effect; it's the technique often used on modern robot toys to create an animated face.

Next month

We will get colourful next month and build a remote-controlled mood light. It uses a low-cost RaspberryPi HAT with eight super-bright, individually controllable, RGB LEDs (APA102). The code for this will demonstrate MMBASIC's SPI commands.

Questions? Please email Phil at:
contactus@micromite.org

PRACTICAL ELECTRONICS IS PLEASED TO OFFER YOU THESE ELECTRONICS CD-ROMS



GCSE ELECTRONICS

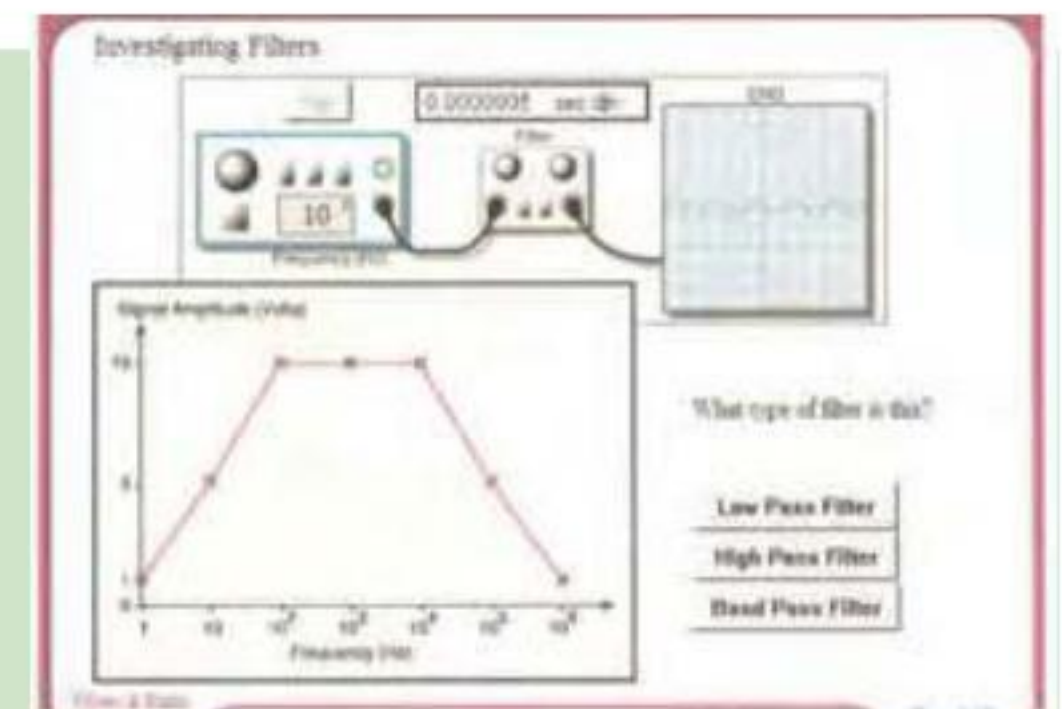
Suitable for any student who is serious about studying and who wants to achieve the best grade possible. Each program's clear, patient and structured delivery will aid understanding of electronics and assist in developing a confident approach to answering GCSE questions. The CD-ROM will be invaluable to anyone studying electronics, not just GCSE students.

- * Contains comprehensive teaching material to cover the National Curriculum syllabus
- * Regular exercises reinforce the teaching points
- * Retains student interest with high quality animation and graphics
- * Stimulates learning through interactive exercises
- * Provides sample examination questions with model solutions
- * Authored by practising teachers
- * Covers all UK examination board syllabuses
- * Caters for all levels of ability
- * Useful for self-tuition and revision

SUBJECTS COVERED

Electric Circuits – Logic Gates – Capacitors & Inductors – Relays – Transistors – Electric Transducers – Operational Amplifiers – Radio Circuits – Test Instruments

Over 100 different sections under the above headings



£12.50
inc. VAT and P&P

CIRCUIT WIZARD

Circuit Wizard is a revolutionary software system that combines circuit design, PCB design, simulation and CAD/CAM manufacture in one complete package. Two versions are available – Standard or Professional.

By integrating the entire design process, Circuit Wizard provides you with all the tools necessary to produce an electronics project from start to finish – even including on-screen testing of the PCB prior to construction!

- * Circuit diagram design with component library (500 components Standard, 1500 components Professional)
- * Virtual instruments (4 Standard, 7 professional)
- * On-screen animation
- * Interactive circuit diagram simulation
- * True analogue/digital simulation
- * Simulation of component destruction
- * PCB Layout
- * Interactive PCB layout simulation
- * Automatic PCB routing
- * Gerber export
- * Multi-level zoom (25% to 1000%)
- * Multiple undo and redo
- * Copy and paste to other software
- * Multiple document support

This software can be used with the *Jump Start* and *Teach-In 2011* series (and the *Teach-In 4* book).

Standard **£61.25** inc. VAT. Professional **£75** plus VAT.

**From
£49.00**

TINA Design Suite V12

Analogue, Digital, Symbolic, RF, MCU and Mixed-Mode
Circuit Simulation and PCB Design with TINA

TINA Design Suite V12 is a powerful yet affordable software package for analysing, designing and real time testing analogue, digital, MCU, and mixed electronic circuits and their PCB layouts. You can also analyse RF, communication, optoelectronic circuits, test and debug microcontroller applications.

Enter and analyse any circuit up to 100 nodes (student), or 200 with the Basic (Hobbyist) version within minutes with TINA's easy-to-use schematic editor. Enhance your schematics by adding text and graphics. Choose components from the large library containing more than 10,000 manufacturer models. Analyse your circuit through more than 20 different analysis modes or with 10 high tech virtual instruments.

Present your results in TINA's sophisticated diagram windows, on virtual instruments, or in the live interactive mode where you can even edit your circuit during operation.

Customise presentations using TINA's advanced drawing tools to control text, fonts, axes, line width, colour and layout. You can create and print documents directly inside TINA or cut and paste your results into your favourite word processing or DTP package.

TINA includes the following Virtual Instruments: Oscilloscope, Function Generator, Multimeter, Signal Analyser/Bode Plotter, Network Analyser, Spectrum Analyser, Logic Analyser, Digital Signal Generator, XY Recorder.

This offer gives you a CD-ROM – the software will need registering (FREE) with Designsoft (TINA), details are given within the package.

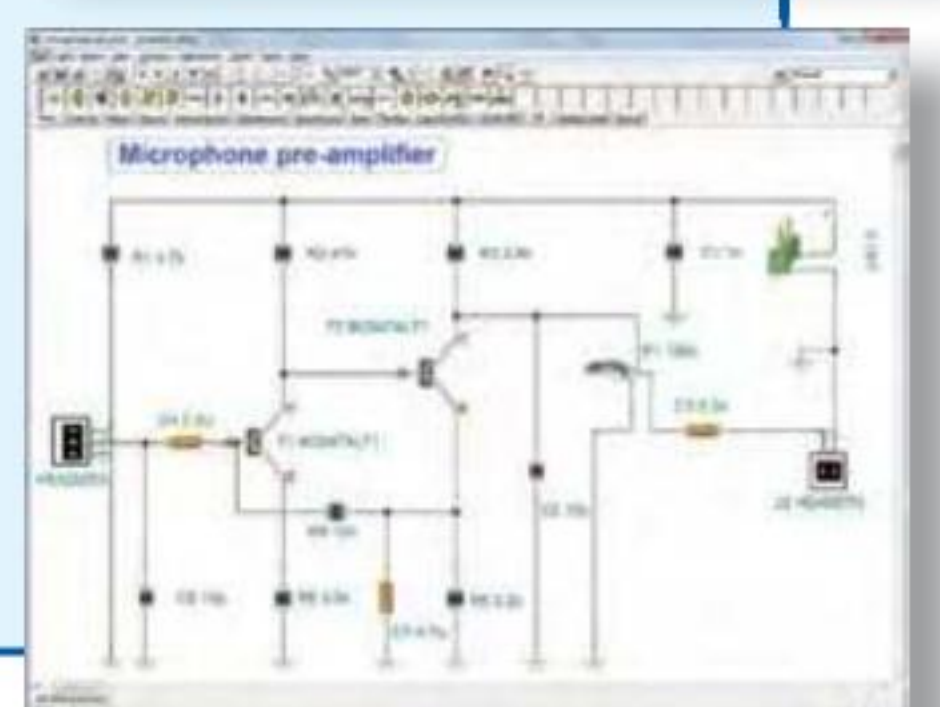
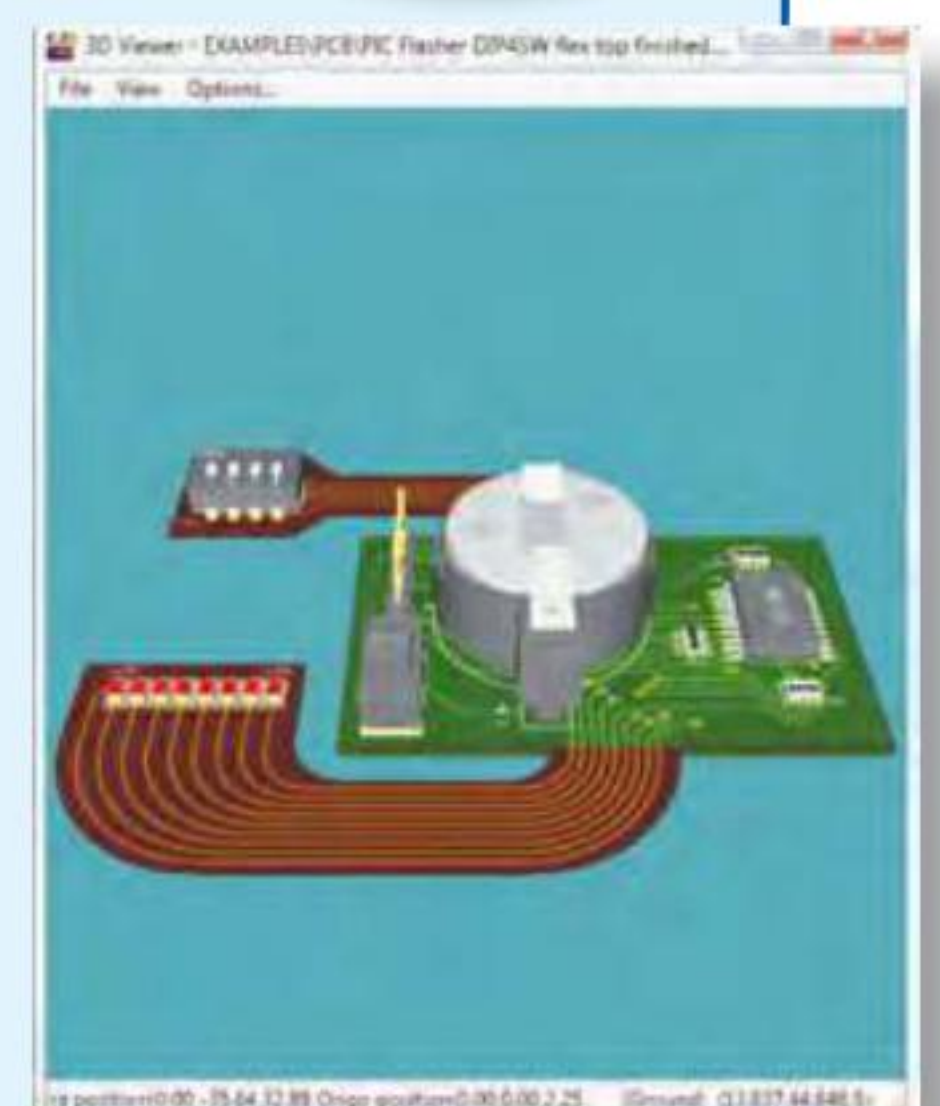
Get TINA 12 Design Suite (Hobbyist) for £129 or Student V12 version for £49

Prices include VAT and UK postage

+ get a 1 year free subscription for TINACloud the breakthrough cloud version of TINA which you can run on most operating systems and computers, including PCs, Macs, thin clients iPads and other tablets – even on many smart phones, smart TVs and e-book readers.

To order please either fill out and return the order form, or call us on **01202 880299**

Alternatively you can order via our secure online shop at: **www.epemag.com**



PICmicro TUTORIALS AND PROGRAMMING

HARDWARE

PICmicro Multiprogrammer Board and Development Board

Suitable for use with the three software packages listed below

This flexible PICmicro microcontroller programmer board and combination board allows students and professional engineers to learn how to program PICmicro microcontrollers as well as program a range of 8, 18, 28 and 40 pin devices from the 12, 16 and 18 series PICmicro ranges. For those who want to learn, choose one or all of the packages below to use with the hardware.

- Makes it easier to develop PICmicro projects
- Supports low cost Flash-programmable PICmicro devices
- Fully featured integrated displays – 16 individual LEDs, quad 7-segment display and alphanumeric LCD display
- Supports PICmicro microcontrollers with A/D converters
- Fully protected expansion bus for project work
- USB programmable
- Compatible with the E-blocks range of accessories



£159.99 including VAT and postage (worldwide)

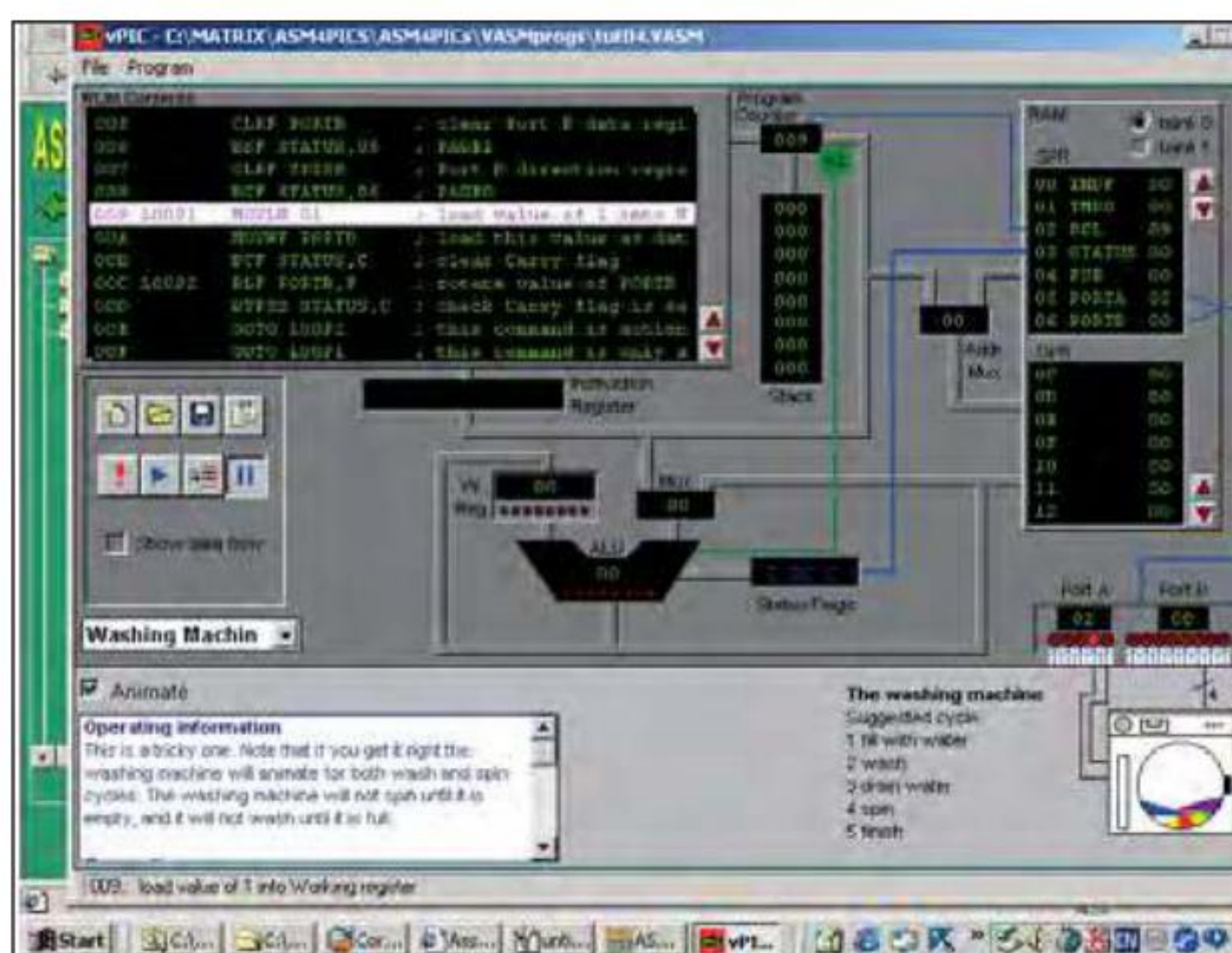
SOFTWARE

ASSEMBLY FOR PICmicro V7 (Formerly PICTutor)

Assembly for PICmicro microcontrollers V7.0 (previously known as PICTutor) by John Becker contains a complete course in programming the PIC16F84, 16F88 and 16F877a PICmicro microcontroller from Arizona Microchip. It starts with fundamental concepts and extends up to complex programs including watchdog timers, interrupts and sleep modes.

The CD makes use of the latest simulation techniques which provide a superb tool for learning: the Virtual PICmicro microcontroller, this is a simulation tool that allows users to write and execute MPASM assembler code for the PIC16F84 microcontroller on-screen. Using this you can actually see what happens inside the PICmicro MCU as each instruction is executed, which enhances understanding.

- Comprehensive instruction through 45 tutorial sections
- Includes Vlab, a Virtual PICmicro microcontroller: a fully functioning simulator
- Tests, exercises and projects covering a wide range of PICmicro MCU applications
- Includes MPLAB assembler
- Visual representation of a PICmicro showing architecture and functions
- Expert system for code entry helps first time users
- Shows data flow and fetch execute cycle and has challenges (washing machine, lift, crossroads etc.)
- Imports MPASM files.

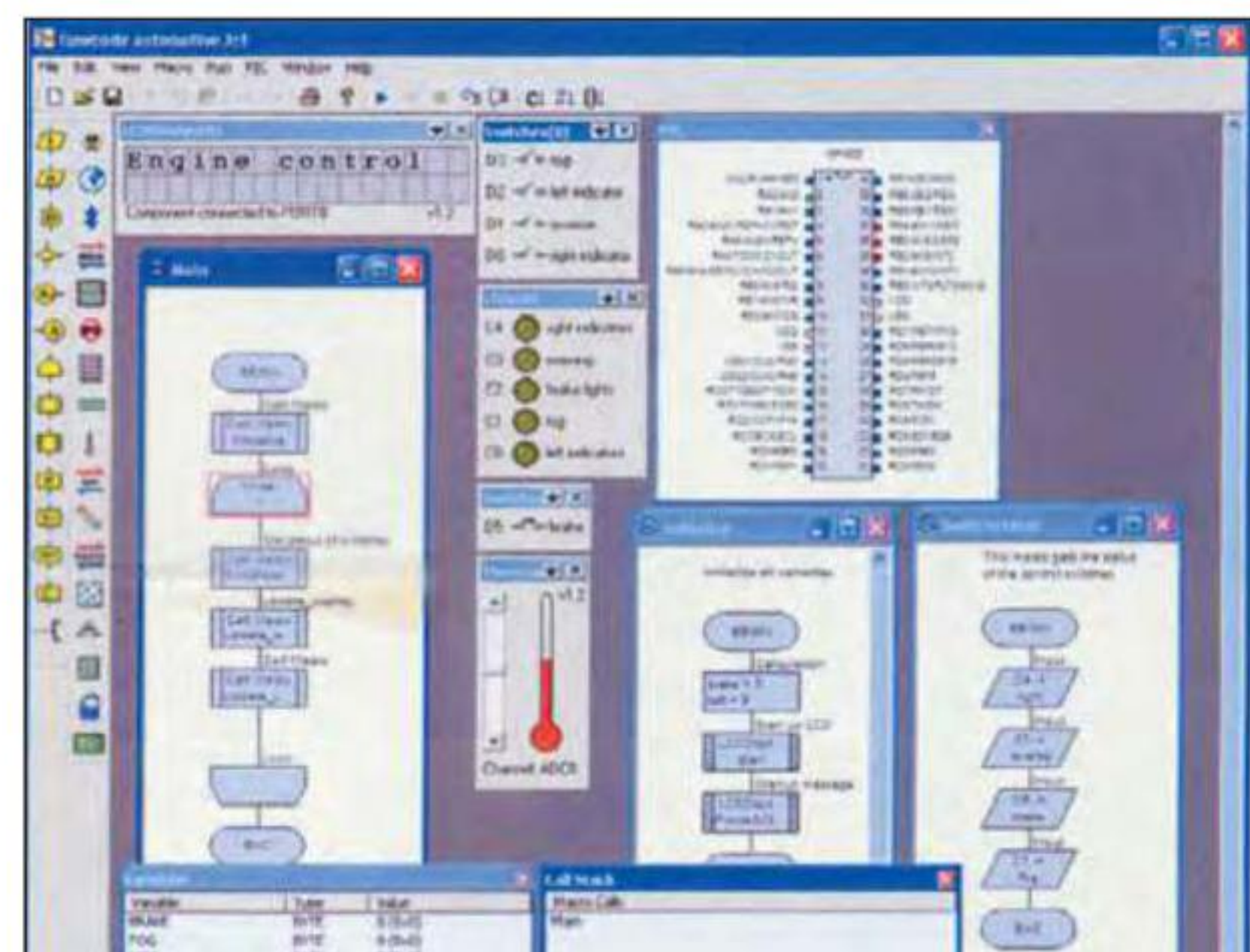


FLOWCODE FOR PICmicro V8

Flowcode is a very high level language programming system based on flowcharts. Flowcode allows you to design and simulate complex systems in a matter of minutes. A powerful language that uses macros to facilitate the control of devices like 7-segment displays, motor controllers and LCDs. The use of macros allows you to control these devices without getting bogged down in understanding the programming. When used in conjunction with the development board this provides a seamless solution that allows you to program chips in minutes.

- Requires no programming experience
- Allows complex PICmicro applications to be designed quickly
- Uses international standard flow chart symbols
- Full on-screen simulation allows debugging and speeds up the development process.
- Facilitates learning via a full suite of demonstration tutorials
- Produces code for a wide range of devices
- 16-bit arithmetic strings and string manipulation
- Pulse width modulation
- I2C.

Please note: Due to popular demand, Flowcode is now available as a download. Please include your email address and a username (of your choice) on your order. A unique download code will then be emailed to you.



PRICES

Prices for each of the CD-ROMs above are:
(Order form on next page)

(UK and EU customers add VAT to 'plus VAT' prices)

Single License £99 plus VAT

Site Licence £499 plus VAT

Flowcode Contact us for pricing
(choose PIC-8b, PIC-16b, PIC-32b, AVR/Arduino, ARM)

ELECTRONICS TEACH-IN 2

ELECTRONICS TEACH-IN 2 CD-ROM USING PIC MICROCONTROLLERS A PRACTICAL INTRODUCTION

This *Teach-In* series of articles was originally published in *EPE* in 2008 and, following demand from readers, has been collected together in the *Electronics Teach-In 2* CD-ROM.

The series is aimed at those using PIC microcontrollers for the first time. Each part of the series includes breadboard layouts to aid understanding and a simple programmer project is provided.

Also included are 29 *PIC N' Mix* articles, also republished from *EPE*. These provide a host of practical programming and interfacing information, mainly for those that have already got to grips with using PIC microcontrollers. An extra four part beginners guide to using the C programming language for PIC microcontrollers is also included.

The CD-ROM also contains all of the software for the *Teach-In 2* series and *PIC N' Mix* articles, plus a range of items from Microchip – the manufacturers of the PIC microcontrollers. The material has been compiled by Wimborne Publishing Ltd. with the assistance of Microchip Technology Inc.

CD-ROM

Order code ETI2 CD-ROM

£9.50

ELECTRONICS TEACH-IN 3

ELECTRONICS TEACH-IN 3 CD-ROM

The three sections of this CD-ROM cover a very wide range of subjects that will interest everyone involved in electronics, from hobbyists and students to professionals. The first 80-odd pages of *Teach-In 3* are dedicated to *Circuit Surgery*, the regular *EPE* clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers the practical aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and identifying components, are covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by the readers of *EPE*.

The CD-ROM also contains the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge, TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The *Teach-In 1* series covers everything from Electric Current through to Microprocessors and Microcontrollers and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

CD-ROM

Order code ETI3 CD-ROM

£8.50

ELECTRONICS TEACH-IN 4

ELECTRONICS TEACH-IN 4 CD-ROM A BROAD-BASED INTRODUCTION TO ELECTRONICS.

The *Teach-In 4* CD-ROM covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the Diploma in Engineering, Level 2.

The CD-ROM also contains the full *Modern Electronics Manual*, worth £29.95. The Manual contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to all those interested in learning about electronics or brushing up on their theory, be they hobbyists, students or professionals.

CD-ROM

Order code ETI4 CD-ROM £8.99

ELECTRONICS TEACH-IN 5

ELECTRONICS TEACH-IN BUNDLE – FOR PARTS 3, 4 & 5

CD-ROM

Order code ETIB3 CD-ROM

£18.95

ORDERING ALL PRICES INCLUDE UK POSTAGE

Standard/Student/Basic (Hobbyist)
Version price includes postage to most countries
in the world EU residents outside the UK add £5 for
airmail postage per order



Please send me: CD-ROM ORDER FORM

- ☐ Assembly for PICmicro V6
☐ 'C' for 16 Series PICmicro V6

Version required:
☐ Single Licence
☐ Site licence

Note: The software on each version is the same, only the licence for use varies.

- ☐ PICmicro Multiprogrammer Board and Development Board (hardware)

- ☐ Circuit Wizard – Standard
☐ Circuit Wizard – Professional

- ☐ GCSE Electronics

- ☐ TINA Design Suite V11 Basic (Hobbyist)

- ☐ TINA Design Suite V11 (Student)

- ☐ Teach-In 2
☐ Teach-In 3
☐ Teach-In 4
☐ Teach-In Bundle
(3, 4 and 5)

Full name

Address

..... Post code Tel

Email:

☐ I enclose cheque/PO in £ sterling payable to Practical Electronics for £

☐ Please charge my Visa/Mastercard: £

Valid From: Card expiry date:

Card No:

Card Security Code

(The last 3 digits on or just under the signature strip)



ELECTRONICS TEACH-IN 5 JUMP START

FREE
CD-ROM

15 design and build circuit projects dedicated to newcomers or those following courses in school and colleges. The projects are: Moisture Detector, Quiz Machine, Battery Voltage Checker, Solar-Powered Charger, Versatile Theft Alarm, Spooky Circuits, Frost Alarm, Mini Christmas Lights, iPod Speaker, Logic Probe, DC Motor Controller, Egg Timer, Signal Injector Probe, Simple Radio Receiver, Temperature Alarm.

PLUS:

PIC N MIX – starting out with PIC Microcontrollers and PRACTICALLY SPEAKING – the techniques of project construction.

FREE CD-ROM – The free CD-ROM is the complete *Teach-In 2* book providing a practical introduction to PIC Microprocessors plus MikroElektronika, Microchip and L-Tek PoScope software.

160 Pages

Order code ETI5

£8.99

Single License and Site License Versions

– overseas readers add £5 to the basic price of each order for airmail postage (**do not add VAT** unless you live in an EU (European Union) country, then add VAT at 20% or provide your official VAT registration number).

Send your order to:
Electron Publishing Limited
113 Lynwood Drive
Wimborne
Dorset BH21 1UU

To order by phone call: **01202 880299**

Goods are normally sent within seven days

E-mail: stewart.kearn@wimborne.co.uk

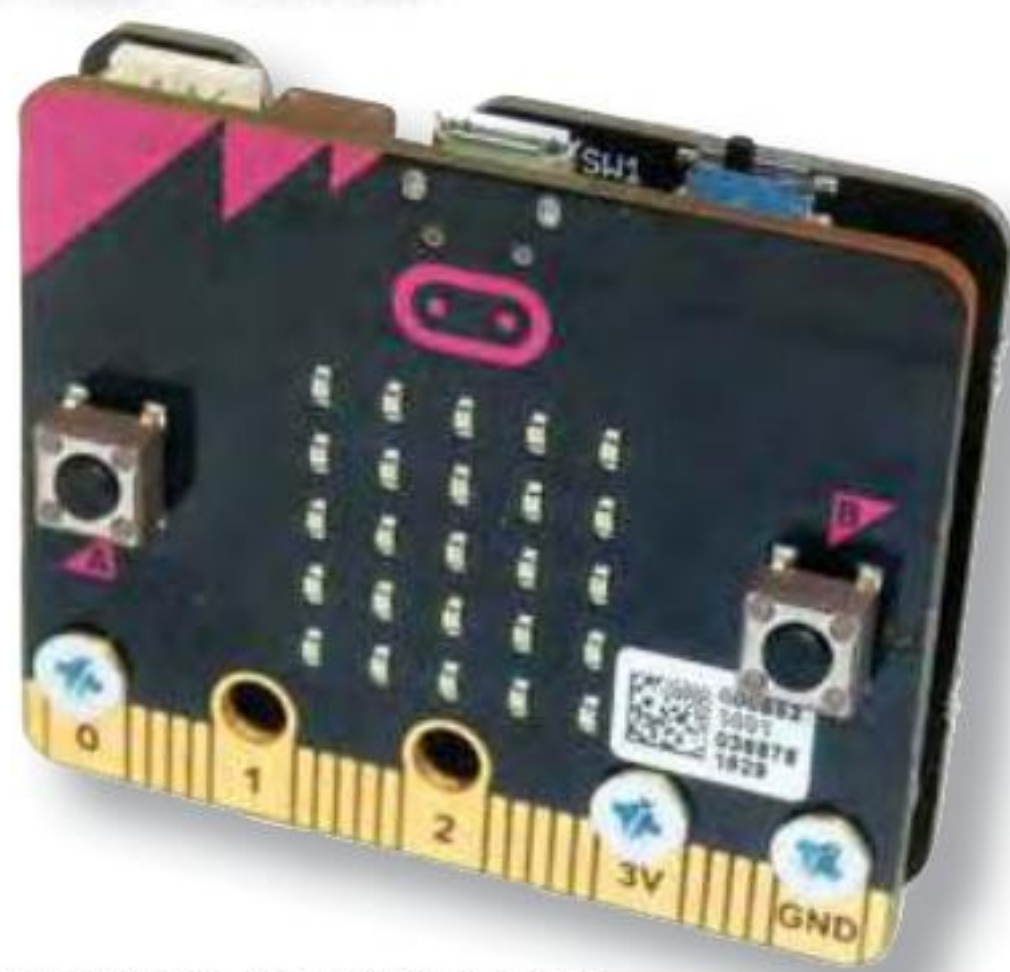
Online shop:

www.epemag.com

DIRECT BOOK SERVICE

Teach-In 2017

Introducing the BBC micro:bit



PYTHON CODING ON THE BBC MICRO:BIT

Jim Gatenby

Python is the leading programming language, easy to learn and widely used by professional programmers. This book uses MicroPython, a version of Python adapted for the BBC Micro:bit.

Among the many topics covered are: main features of the BBC micro:bit including a simulation in a web browser screen; various levels of programming languages; Mu Editor for writing, saving and retrieving programs, with sample programs and practice exercises; REPL, an interactive program for quickly testing lines of code; scrolling messages, creating and animating images on the micro:bit's LEDs; playing and creating music, sounds and synthesized speech; using the on-board accelerometer to detect movement of the micro:bit on three axes; glossary of computing terms.

This book is written using plain English, avoids technical jargon wherever possible and covers many of the coding instructions and methods which are common to most programming languages. It should be helpful to beginners of any age, whether planning a career in computing or writing code as an enjoyable hobby.

118 Pages

Order code PYTH MBIT

£7.99



GETTING STARTED WITH THE BBC MICRO:BIT

Mike Tooley

Not just an educational resource for teaching youngsters coding, the BBC micro:bit is a tiny low cost, low-profile ARM-based single-board computer. The board measures 43mm x 52mm but despite its diminutive footprint it has all the features of a fully fledged microcontroller together with a simple LED matrix display, two buttons, an accelerometer and a magnetometer.

Mike Tooley's book will show you how the micro:bit can be used in a wide range of applications from simple domestic gadgets to more complex control systems such as those used for lighting, central heating and security applications. Using Microsoft Code Blocks, the book provides a progressive introduction to coding as well as interfacing with sensors and transducers.

Each chapter concludes with a simple practical project that puts into practice what the reader has learned. The featured projects include an electronic direction finder, frost alarm, reaction tester, battery checker, thermostatic controller and a passive infrared (PIR) security alarm.

No previous coding experience is assumed, making this book ideal for complete beginners as well as those with some previous knowledge. Self-test questions are provided at the end of each chapter, together with answers at the end of the book. So whatever your starting point, this book will take you further along the road to developing and coding your own real-world applications.

108 Pages

Order code BBC MBIT

£7.99

THEORY AND REFERENCE

PRACTICAL ELECTRONICS HANDBOOK – 6th Ed

Ian Sinclair

440 pages

Order code NE21

£33.99

STARTING ELECTRONICS – 4th Ed

Keith Brindley

296 pages

Order code ELSEV100

£18.99

ELECTRONIC CIRCUITS – FUNDAMENTALS & APPLICATIONS – Updated version

Mike Tooley

400 pages

Order code TF43

£32.99

FUNDAMENTAL ELECTRICAL AND ELECTRONIC PRINCIPLES – 3rd Ed

C.R. Robertson

368 pages

Order code TF47

£21.99

A BEGINNER'S GUIDE TO TTL DIGITAL ICs

Robert Penfold

142 pages

OUT OF PRINT BP332

£5.45

UNDERSTANDING ELECTRONIC CONTROL SYSTEMS

Owen Bishop

228 pages

Order code NE35

£36.99

The books listed here have been selected by the *Practical Electronics* editorial staff as being of special interest to everyone involved in electronics and computing. They are supplied by mail order direct to your door.

FOR A FULL DESCRIPTION OF THESE BOOKS AND CD-ROMS SEE THE SHOP ON OUR WEBSITE

www.epemag.com

All prices include UK postage

MICROPROCESSORS

INTERFACING PIC MICROCONTROLLERS – 2nd Ed

Martin Bates

298 pages

Order code NE48

£34.99

PROGRAMMING 16-BIT PIC MICROCONTROLLERS IN C – LEARNING TO FLY THE PIC24

Lucio Di Jasio (Application Segments Manager, Microchip, USA)

496 pages + CD-ROM

Order code NE45

£38.00

INTRODUCTION TO MICROPROCESSORS AND MICROCONTROLLERS – 2nd Ed

John Crisp

222 pages

Order code NE31

£29.99

THE PIC MICROCONTROLLER YOUR PERSONAL INTRODUCTORY COURSE – 3rd Ed

John Morton

270 pages

Order code NE36

£25.00

PIC IN PRACTICE – 2nd Ed

David W. Smith

308 pages

Order code NE39

£24.99

MICROCONTROLLER COOKBOOK

Mike James

240 pages

Order code NE26

£36.99

BOOK ORDERING DETAILS

All prices include UK postage.

For postage, add £3 per book to Europe, £4 for rest of the world per book.

CD-ROM prices include VAT and/or postage to anywhere in the world.

Send a cheque, (£ sterling only) made payable to: **Practical Electronics** or credit card details (Visa or Mastercard) to:

**Electron Publishing Limited,
113 Lynwood Drive, Wimborne, Dorset BH21 1UU**

Books are normally sent within seven days of receipt of order.

Please check price (see latest issue of Practical Electronics or website) before ordering from old lists.

For a full description of these books please see the shop on our website.

Tel: 01202 880299 – E-mail: stewart.kearn@wimborne.co.uk

Order from our online shop at: www.epemag.com



ARDUINO

Teach-In 2016

See opposite for our popular introduction to the Arduino



ARDUINO FOR DUMMIES John Nussey

Arduino is no ordinary circuit board. Whether you're an artist, a designer, a programmer, or a hobbyist, Arduino lets you learn about and play with electronics. You'll discover how to build a variety of circuits that can sense or control real-world objects, prototype your own product, and even create interactive artwork. This handy guide is exactly what you need to build your own Arduino project – what you make is up to you!

- **Learn by doing** – start building circuits and programming your Arduino with a few easy examples – right away!
- **Easy does it** – work through Arduino sketches line by line, and learn how they work and how to write your own.
- **Solder on!** – don't know a soldering iron from a curling iron? No problem! You'll learn the basics and be prototyping in no time.
- **Kitted out** – discover new and interesting hardware to turn your Arduino into anything from a mobile phone to a Geiger counter.
- **Become an Arduino savant** – find out about functions, arrays, libraries, shields and other tools that let you take your Arduino project to the next level
- **Get social** – teach your Arduino to communicate with software running on a computer to link the physical world with the virtual world

438 Pages **Order code ARDDUM01** £19.99



EXPLORING ARDUINO Jeremy Blum

Arduino can take you anywhere. This book is the roadmap. Exploring Arduino shows how to use the world's most popular microcontroller to create cool, practical, artistic and educational projects. Through lessons in electrical engineering, programming and human-computer interaction, this book walks you through specific, increasingly complex projects, all the while providing best practices that you can apply to your own projects once you've mastered these. You'll acquire valuable skills – and have a whole lot of fun.

- **Explore** the features of commonly used Arduino boards
- **Use** Arduino to control simple tasks or complex electronics
- **Learn** principles of system design, programming and electrical engineering
- **Discover** code snippets, best practices and system schematics you can apply to your original projects
- **Master** skills you can use for engineering endeavours in other fields and with different platforms

357 Pages **Order code EXPARD01** £26.99

COMPUTING AND ROBOTICS

NEWNES INTERFACING COMPANION Tony Fischer-Cripps

295 pages **Order code NE38** £41.00

HOW TO BUILD A COMPUTER MADE EASY Robert Penfold

120 pages **Order code BP707** £8.49

EASY PC CASE MODDING Robert Penfold

192 pages + CD-ROM **Order code BP542** £8.99

FREE DOWNLOADS TO PEP-UP AND PROTECT YOUR PC Robert Penfold

128 pages **Order code BP722** £7.99

WINDOWS XP EXPLAINED N. Kantaris and P.R.M. Oliver

264 pages **Order code BP514** £7.99

THE INTERNET – TWEAKS, TIPS AND TRICKS Robert Penfold

128 pages **Order code BP721** £7.99

eBAY – TWEAKS, TIPS AND TRICKS Robert Penfold

128 pages **Order code BP716** £7.50

AN INTRODUCTION TO eBAY FOR THE OLDER GENERATION Cherry Nixon

120 pages **Order code BP709** £8.49

HOW TO FIX YOUR PC PROBLEMS Robert Penfold

128 pages **Order code BP705** £8.49

AN INTRODUCTION TO WINDOWS VISTA P.R.M. Oliver and N. Kantarris

120 pages **Order code BP703** £8.49

WINDOWS 8.1 EXPLAINED

180 Pages **Order code BP747** £10.99

AN INTRODUCTION TO THE NEXUS 7

118 Pages **Order code BP744** £8.99

COMPUTING FOR THE OLDER GENERATION Jim Gatenby

308 pages **Order code BP601** £8.99

ANDROIDS, ROBOTS AND ANIMATRONICS Second Edition – John Iovine

224 pages **Order code MGH1** £16.99

ROBOT BUILDERS COOKBOOK Owen Bishop

366 pages **Order code NE46** £26.00

INTRODUCING ROBOTICS WITH LEGO MINDSTORMS Robert Penfold

288 pages + **Order code BP901** £14.99

MORE ADVANCED ROBOTICS WITH LEGO MINDSTORMS Robert Penfold

298 pages **Order code BP902** £14.99

HOW TO FIX YOUR PC PROBLEMS Robert Penfold

128 pages **Order code BP705** £8.49

WINDOWS 7 – TWEAKS, TIPS AND TRICKS Andrew Edney

120 pages **Order code BP708** £8.49

GETTING STARTED IN COMPUTING FOR THE OLDER GENERATION Jim Gatenby

120 pages **Order code BP704** £8.49

WINDOWS 8.1 EXPLAINED Noel Kantaris

180 Pages **Order code BP747** £10.99

COMPUTING WITH A LAPTOP FOR THE OLDER GENERATION Robert Penfold

120 pages **Order code BP702** £8.49

AN INTRODUCTION TO EXCEL SPREADSHEETS Jim Gatenby

18 pages **Order code BP701** £8.49

KINDLE FIRE HDX EXPLAINED

118 Pages **Order code BP743** £8.99

AUDIO & VIDEO

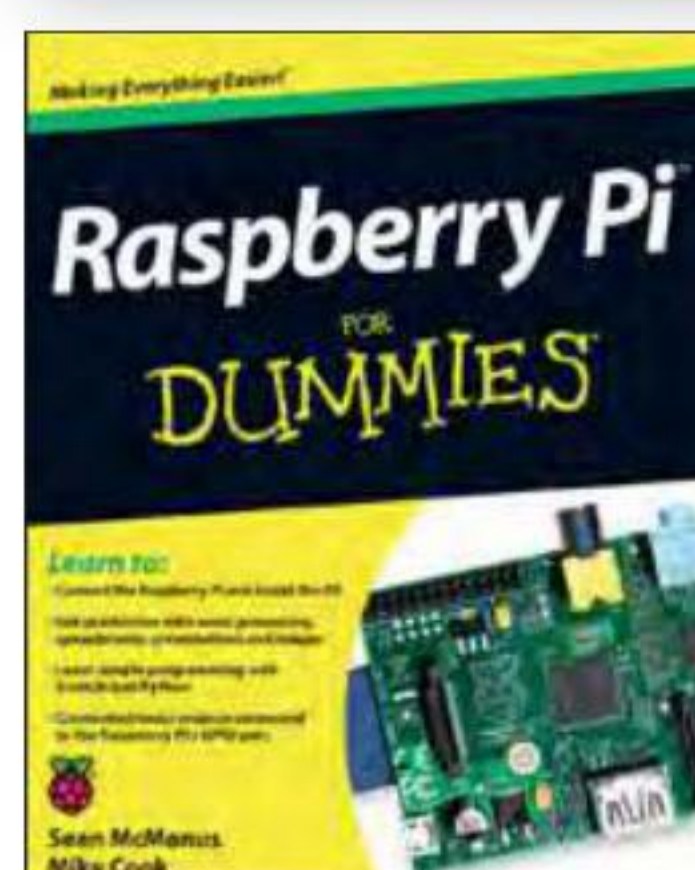
VALVE AMPLIFIERS – 4th Ed Morgan Jones

288 pages **Order code ELSEV33** £46.99

BUILDING VALVE AMPLIFIERS Morgan Jones

368 pages **Order code NE40** £29.00

RASPBERRY PI



RASPBERRY Pi FOR DUMMIES Sean McManus and Mike Cook

Write games, compose and play music, even explore electronics – it's easy as Pi! The Raspberry Pi offers a plethora of opportunities, and this great resource guides you step-by-step, from downloading, copying, and installing the software to learning about Linux and finding cool new programs for work, photo editing, and music. You'll discover how to write your own Raspberry Pi programs, create fun games, and much more!

Open this book and find:

What you can do with Python; Ways to use the Raspberry Pi as a productivity tool; How to surf the web and manage files; Secrets of Sonic Pi music programming; A guide to creating animations and arcade games; Fun electronic games you can build; How to build a 3D maze in Minecraft; How to play music and videos on your Raspberry Pi.

400 Pages **Order code RPiDUM01** £17.99

RASPBERRY Pi MANUAL: A practical guide to the revolutionary small computer

176 pages **Order code H001** £17.99

RASPBERRY Pi USER-GUIDE – 4th Ed

262 pages **Order code JW001** £20.90

PROGRAMMING THE RASPBERRY Pi

192 pages **Order code MGH4** £10.99

GETTING STARTED WITH RASPBERRY Pi

164 pages **Order code OR01** £11.50

TEACH-IN BOOKS

ELECTRONICS TEACH-IN 6



ONLY AVAILABLE ON CD-ROM

ELECTRONICS TEACH-IN 6 A COMPREHENSIVE GUIDE TO RASPBERRY Pi Mike & Richard Tooley

Teach-In 6 contains an exciting series of articles that provides a complete introduction to the Raspberry Pi, the low-cost computer that has taken the education and computing world by storm.

This latest book in our Teach-In series will appeal to electronic enthusiasts and computer buffs wanting to get to grips with the Raspberry Pi.

Anyone considering what to do with their Pi, or maybe they have an idea for a project but don't know how to turn it into reality, will find Teach-In 6 invaluable. It covers: Programming, Hardware, Communications, Pi Projects, Pi Class, Python Quickstart, Pi World, and Home Baking.

The CD-ROM also contains all the necessary software for the series so that readers can get started quickly and easily with the projects and ideas covered.

160 Pages

Order code ETI6

£8.99

ELECTRONICS TEACH-IN 7 (Includes free CD-ROM)



RUNNING LOW – ORDER NOW!

ELECTRONICS TEACH-IN 7 DISCRETE LINEAR CIRCUIT DESIGN Mike & Richard Tooley

Teach-In 7 is a complete introduction to the design of analogue electronic circuits. Ideal for everyone interested in electronics as a hobby and for those studying technology at schools and colleges. Supplied with a free cover-mounted CD-ROM containing all the circuit software for the course, plus demo CAD software for use with the Teach-In series.

Discrete Linear Circuit Design* Understand linear circuit design* Learn with 'TINA' – modern CAD software* Design simple, but elegant circuits* Five projects to build: Pre-amp, Headphone Amp, Tone Control, VU-meter, High Performance Audio Power Amp.

PLUS

Audio Out – an analogue expert's take on specialist circuits
Practically Speaking – the techniques of project building

160 Pages

Order code ETI7

£8.99

ELECTRONICS TEACH-IN 8



ONLY AVAILABLE ON CD-ROM

ELECTRONICS TEACH-IN 8 INTRODUCING THE ARDUINO Mike & Richard Tooley

Hardware – learn about components and circuits; Programming – powerful integrated development system; Microcontrollers – understand control operations; Communications – connect to PCs and other Arduinos

This exciting series has been designed for electronics enthusiasts who want to get to grips with the inexpensive, immensely popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. Teach-In 8 will provide a one-stop source of ideas and practical information.

The Arduino offers a remarkably effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle through wireless or the Internet. Teach-In 8 is based around a series of practical projects with plenty of information to customise each project.

This book also includes *PIC n' Mix*: PICs and the PICKit 3 - A Beginners guide by Mike O'Keefe and *Circuit Surgery* by Ian Bell - State Machines part 1 and 2.

The CD-ROM includes files for Teach-In 8 plus Microchip MPLAB IDE XC8 8-bit Compiler and PICKit 3 User Guide. Also included is Lab-Nation Smartscope software.

160 Pages

Order code ETI8

£8.99

**VISIT OUR WEBSITE FOR MORE BOOKS
AND FAST, EASY ONLINE ORDERING
WWW.EPEMAG.COM**

BOOK ORDER FORM

Full name:

Address:

Post code: Telephone No:

Email:

☐ I enclose cheque/PO payable to Practical Electronics for £

☐ Please charge my card £ Card expiry date:

Card Number

Valid From Date Card Security Code (The last three digits on or just below the signature strip)

Please send book order codes:

.....

Please continue on separate sheet of paper if necessary

THE BASIC SOLDERING GUIDE

**LEARN TO SOLDER
SUCCESSFULLY!
ALAN WINSTANLEY**

The No.1 resource for learning all the basic aspects of electronics soldering by hand.

With more than 80 high quality colour photographs, this book explains the correct choice of soldering irons, solder, fluxes and tools. The techniques of how to solder and desolder electronic components are then explained in a clear, friendly and non-technical fashion so you'll be soldering successfully in next to no time! The book also includes sections on reflow soldering and desoldering techniques, potential hazards, useful resources and a very useful troubleshooting guide.

Also ideal for those approaching electronics from other industries, the *Basic Soldering Guide Handbook* is the best resource of its type, and thanks to its excellent colour photography and crystal clear text, the art of soldering can now be learned by everyone!

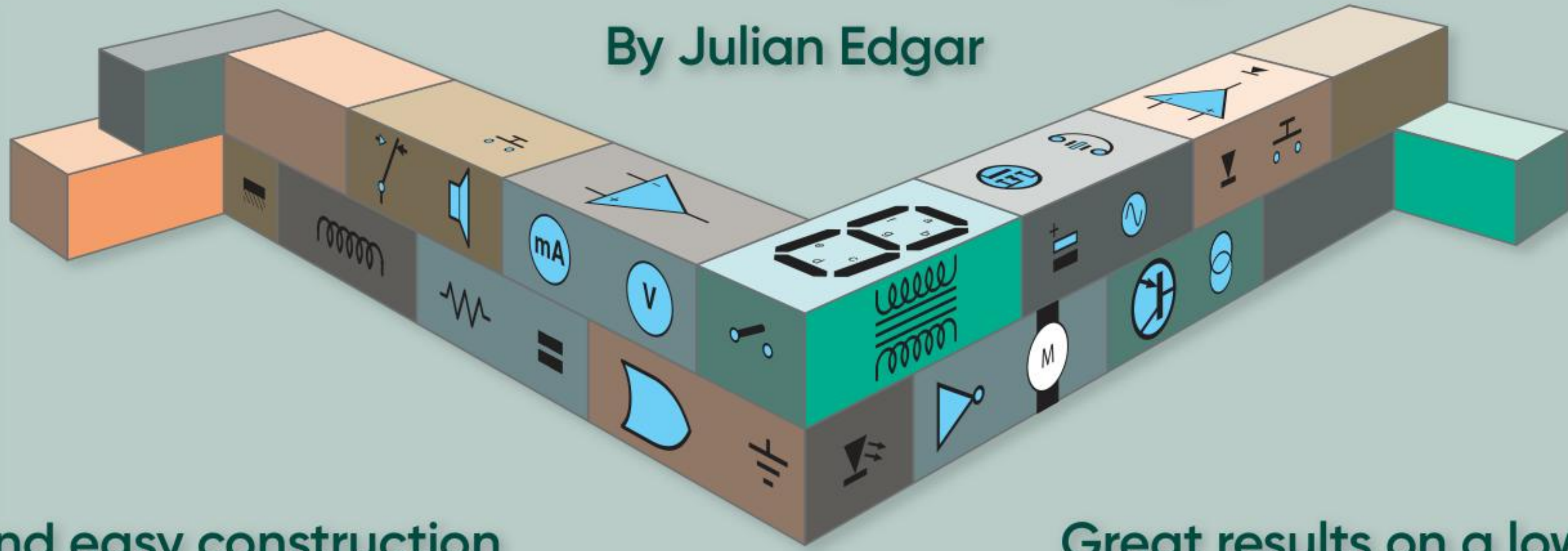
86 Pages

Order code AW1

£9.99

Electronic Building Blocks

By Julian Edgar



Quick and easy construction

Great results on a low budget

Digital mains voltmeter and ammeter display



Fig.1. This small instrument displays both AC voltage and current. The current sensor (current transformer) is shown at left – in use, one conductor is passed through the ring. Accuracy is adequate for general use, and the cost is under £3, delivered!

This handy meter combines an AC voltmeter and ammeter but is cheap enough to use as a ‘bells and whistles’ pilot light on single-phase mains-powered equipment.

The meter can read AC voltages from 50-500V and current from 0-100A. The front display is 29mm square and the unit can be easily installed through a 22mm round hole. The LED digits are about 7mm high, available in red, green, blue, yellow and white variants.

It's available from uk.banggood.com (ID: 1488117) and costs just £2.70, delivered. So it really is cheap enough to use as a fancy on/off light on equipment you're building.

Connections are very simple – mains live and neutral are connected

to screw terminals on the rear of the instrument, and the live supply to the device for which current is being measured is passed through the core of the current sensor, which is a current transformer. The current sensor comes with a plug that connects to a socket on the instrument.

Accuracy? At 240V the display was about 2V in error, and at 10A current the display was about 0.5A in error. Both measurements were made using a recently calibrated, good quality multimeter as the reference. So don't expect laboratory-level accuracy, but measurements accurate to within about 5% – at least for common mains voltages and currents. Also note that the device does not have a fast

response. It takes about two seconds to react to a change in current, so you cannot measure real-time consumption of an audio amplifier or similar.

Making an inline current and voltage measurer

I decided to build the instrument into a box and use it as a general-purpose, in-line, current and voltage-measuring device.

A box $83 \times 58 \times 34\text{mm}$ was selected. It needed to be a sturdy enclosure – you don't want to pick a light-duty box, only to drop it and have it crack open, revealing mains wiring. A 22mm hole saw was used to drill through the front of the box to mount the display. The current sensor was positioned at the end of the display, held in place by the lid of the box that holds it firmly captive when the lid is screwed down.



Fig.2. I decided to build the display into a box and use it as an in-line current and voltage-measuring device. The plug and socket were obtained by buying a short extension cable – obviously, the type of plug and socket are country-specific – mine are Australian. The box has a clear plastic lid.



Fig.3. Inside the box, the inlet and outlet power cables can be seen securely anchored. The live wire is passed through the current-measuring core positioned at the end of the display. The current sensor connects to the display via a provided plug and socket. The current sensor is securely held in place when the lid is screwed down.

To provide the inline plug and socket, I selected a short (and so cheap) extension cable, and cut it in half. The cables were fed through holes drilled in each end of the box, and then anchored in place with clamps. The earth wires from each cable were soldered together. (If you use a metal box, ensure you connect it to these earth wires.) The neutral conductors were joined and fed to one of the power terminals of the display. One live wire was fed through the current sensor, then joined to the other live. A short extension was used to connect this live supply



Fig.4. The kitchen kettle drawing just over 10A – and causing a 10V drop!

to the other power terminal on the display. Note that as they were being made, all connections were covered in heatshrink. The current sensor was then plugged into the display and the lid screwed into place.

Using it

The beauty of the inline display is that it is very easy to use – just plug the device you're measuring into the socket and connect the plug to a power outlet.

One of the first things I noticed when using the display was the voltage drop that occurs in our household wiring. For example, the kitchen kettle, which draws about 10A, also causes a 10V voltage drop. That is, 240V is available at the power outlet with the kettle off, and only 230V with the kettle on! And talking about making coffee, I discovered our coffee maker draws 5A when warming up, and 6.6A when actually pouring the coffee.

So whether you buy the display to build into equipment, or put it in a box to use as a portable measuring tool, there are lots of uses for this small, very low-priced instrument.



Teach-In 8 CD-ROM Exploring the Arduino

This CD-ROM version of the exciting and popular *Teach-In 8* series has been designed for electronics enthusiasts who want to get to grips with the inexpensive, immensely popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. *Teach-In 8* provides a one-stop source of ideas and practical information.

The Arduino offers a remarkably effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle wirelessly or via the Internet. *Teach-In 8* is based around a series of practical projects with plenty of information for customisation. The projects can be combined together in many different ways in order to build more complex systems that can be used to solve a wide variety of home automation and environmental monitoring problems. The series includes topics such as RF technology, wireless networking and remote web access.

PLUS: PICs and the PICkit 3 – A beginners guide

The CD-ROM also includes a bonus – an extra 12-part series based around the popular PIC microcontroller, explaining how to build PIC-based systems.

SOFTWARE

The CD-ROM contains the software for both the *Teach-In 8* and PICkit 3 series.



PRICE
£8.99

Includes P&P to UK if
ordered direct from us

ORDER YOUR COPY TODAY!
JUST CALL 01202 880299 OR VISIT www.epemag.com

Practical Electronics PCB SERVICE

PROJECT	CODE	PRICE
MARCH 2018		
Stationmaster Main Board.....	09103171	£17.75
+ Controller Board	09103172	
Build the SC200 Amplifier Module – Power Supply	01109111	£16.45
APRIL 2018		
Spring Reverberation Unit	01104171	£15.30
DDS Sig Gen Lid	Black	£8.05
DDS Sig Gen Lid	Blue	£7.05
DDS Sig Gen Lid	Clear	£8.05
MAY 2018		
High Performance RF Prescaler.....	04112162	£10.45
Micromite BackPack V2.....	07104171	£10.45
Microbridge.....	24104171	£5.60
JUNE 2018		
High Performance 10-Octave Stereo Graphic Equaliser...	01105171	£15.30
JULY 2018		
Touchscreen Appliance Energy Meter – Part 1	04116061	£17.75
Automotive Sensor Modifier	05111161	£12.88
AUGUST 2018		
Universal Temperature Alarm	03105161	£7.05
Power Supply For Battery-Operated Valve Radios	18108171 18108172 18108173 18108174	£27.50
SEPTEMBER 2018		
3-Way Active Crossover	01108171	£22.60
Ultra-low-voltage Mini LED Flasher.....	16110161	£5.60
OCTOBER 2018		
6GHz+ Touchscreen Frequency Counter	04110171	£12.88
Two 230VAC MainsTimers	10108161 10108162	£12.88
NOVEMBER 2018		
Super-7 AM Radio Receiver	06111171	£27.50
FEBRUARY 2019		
1.5kW Induction Motor Speed Controller.....	10105122	£35.00
MARCH 2019		
10-LED Bargraph Main Board	04101181	£11.25
+Processing Board	04101182	£8.60
APRIL 2019		
Heater Controller	10104181	£14.00
MAY 2019		
2× 12V Battery Balancer	14106181	£5.60
Deluxe Frequency Switch.....	05104181	£10.45
USB Port Protector	07105181	£5.60
JUNE 2019		
Arduino-based LC Meter	04106181	£8.00
USB Flexitimer.....	19106181	£10.45
JULY 2019		
Full-wave 10A Universal Motor Speed Controller.....	10102181	£12.90
Recurring Event Reminder	19107181	£8.00
Temperature Switch Mk2.....	05105181	£10.45

PROJECT	CODE	PRICE
AUGUST 2019		
Brainwave Monitor.....	25108181	£12.90
Super Digital Sound Effects Module.....	01107181	£5.60
Watchdog Alarm	03107181	£8.00
PE Theremin (three boards: pitch, volume, VCA)	PETX0819	£19.50
PE Theremin component pack (see p.56, August 2019) ...	PETY0819	£15.00
OCTOBER 2019		
Programmable GPS-synced Frequency Reference	04107181	£11.50
Digital Command Control Programmer for Decoders.....	09107181	£8.75
Opto-isolated Mains Relay (main board).....	10107181	£11.50
Opto-isolated Mains Relay (2 × terminal extension board)...	10107182	
NOVEMBER 2019		
Tinnitus & Insomnia Killer (Jaycar case – see text)	01110181	£8.75
Tinnitus & Insomnia Killer (Altronics case – see text).....	01110182	£8.75
DECEMBER 2019		
Extremely Sensitive Magnetometer.....	04101011	£16.75
Four-channel High-current DC Fan and Pump Controller...	05108181	£8.75
Useless Box	08111181	£11.50
JANUARY 2020		
Isolated Serial Link	24107181	£8.50

PE/EPE PCB SERVICE

Order Code	Project	Quantity	Price
.....			
.....			
.....			
.....			
.....			

Name

Address

.....

Tel

Email

I enclose payment of £..... (cheque/PO in **£ sterling only**)

payable to: **Practical Electronics**

Card No

Valid From Expiry Date

Card Security No






You can also order PCBs by phone, Fax, Email or via the shop on our website on a secure server: www.epemaq.com

PCBs for most recent *PE/EPE* constructional projects are available. From the July 2013 issue onwards, PCBs with eight-digit codes have silk screen overlays and, where applicable, are double-sided, plated-through hole, with solder mask. They are similar to photos in the project articles. Earlier PCBs are likely to be more basic and may not include silk screen overlay, be single-sided, lack plated-through holes and solder mask.

Always check price and availability in the latest issue or online. A large number of older boards are listed for ordering on our website.

We do not supply kits or components for our projects. For older projects it is important to check the availability of all components before purchasing PCBs.

Back issues of articles are available – see Back Issues page for details.

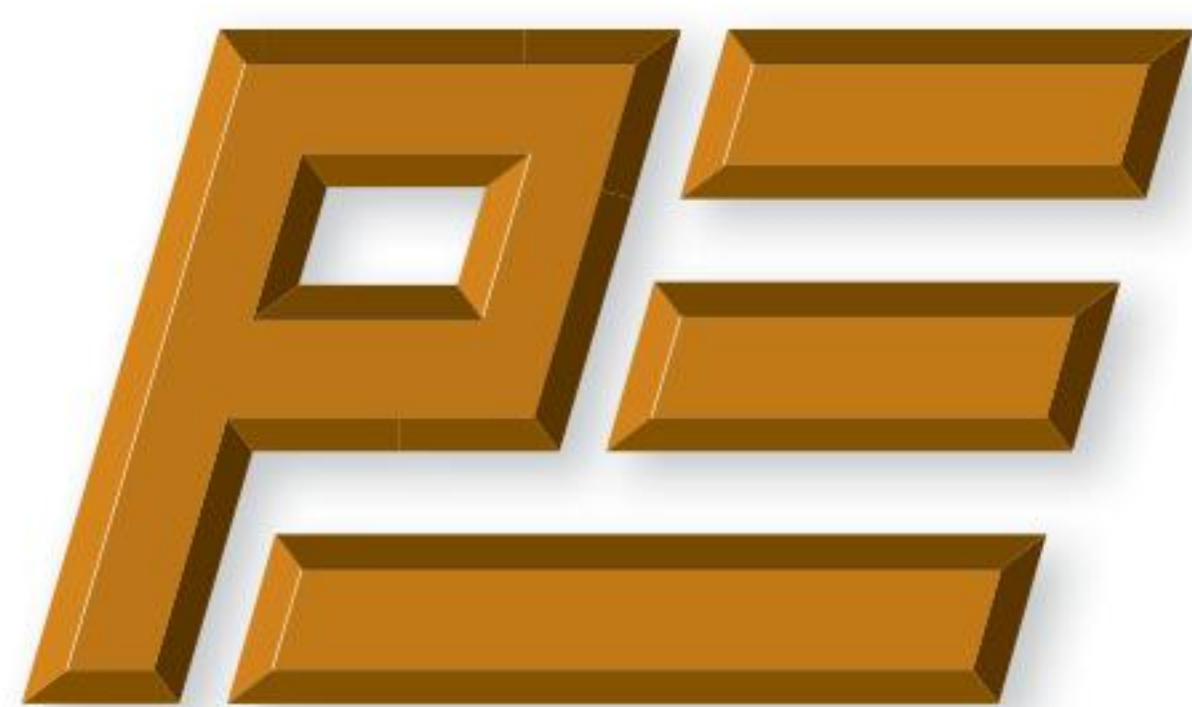
All prices include VAT and UK p&p. Add £3 per board for post to Europe; £4 per board outside Europe. Orders and payment should be sent to:

Practical Electronics, Electron Publishing Ltd
113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU
Tel 01202 880299 Fax 01202 843233
Email: stewart.kearn@epemag.wimborne.co.uk
On-line Shop: www.epemag.com

Cheques should be made payable to 'Practical Electronics' (**Payment in £ sterling only**).

NOTE: While 95% of our boards are held in stock and are dispatched within seven days of receipt of order, please allow a maximum of 28 days for delivery if we need to restock.

CLASSIFIED ADVERTISING



Practical Electronics

If you want your advertisements to be seen by the largest readership at the most economical price then our classified page offers excellent value. The rate for semi-display space is £10 (+VAT) per centimetre high, with a minimum height of 2.5cm. All semi-display adverts have a width of 5.5cm. The prepaid rate for classified adverts is 40p (+VAT) per word (minimum 12 words).

Cheques are made payable to 'Practical Electronics'. **VAT must be added.** Advertisements with remittance should be sent to: Practical Electronics, 113 Lynwood Drive, Wimborne, Dorset, BH21 1UU.

Tel 01202 880299 Email: stewart.kearn@wimborne.co.uk

For rates and further information on display and classified advertising please contact our Advertisement Manager, Stewart Kearns – see below.

Practical Electronics reaches more UK readers than any other UK monthly hobby electronics magazine. Our sales figures prove it.

We have been the leading monthly magazine in this market for the last twenty-seven years.

BOWOOD ELECTRONICS LTD

Suppliers of Electronic Components

www.bowood-electronics.co.uk

Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield, Derbyshire S40 2QR. Sales: 01246 200 222

Send large letter stamp for Catalogue

The British Amateur Electronic Club at:

baec.tripod.com

Has many interesting articles on computers; digital electronics and analogue electronics.



COAST ELECTRONICS

BREAKOUTS-COMPONENTS-
CONTRACT DESIGN-3D PRINTER PARTS-
MUSICAL-MICROCONTROLLERS
WWW.COASTELECTRONICS.CO.UK

MISCELLANEOUS

VALVES AND ALLIED COMPONENTS IN STOCK. Phone for free list. Valves, books and magazines wanted. Geoff Davies (Radio), tel. 01788 574774.

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits. Visit www.cstech.co.uk

Electrical Industries Charity (EIC)

We help people working in the electrical, electronics and energy community as well as their family members and retirees.

We use workplace programmes that give the industry access to financial grants and a comprehensive range of free and confidential services.

www.electricalcharity.org



Andrew Kenny – Qualified Patent Agent
EPO UKIPO USPTO

Circuits Electric Machinery Mechatronics

Web: www.akennypatentm.com

Email: Enquiries@akennypatentm.com

Tel: 0789 606 9725

ADVERTISING INDEX

BRUNNING SOFTWARE	9
CRICKLEWOOD ELECTRONICS	65
ESR ELECTRONIC COMPONENTS	39
HAMMOND ELECTRONICS Ltd	8
JPG ELECTRONICS	80
MICROCHIP	Cover (ii), Cover (iii), 5
PCBWay	44, 58
PEAK ELECTRONIC DESIGN	Cover (iv)
POLABS D.O.O.	61
QUASAR ELECTRONICS	2
SOUNDTRONICS	3
STEWART OF READING	3
TAG-CONNECT	39
TECHNOBOTS	57

Advertisement offices

Electron Publishing Ltd
113 Lynwood Drive
Merley
Wimborne,
Dorset BH21 1UU

Tel 01202 880299

Fax 01202 843233

Email stewart.kearn@wimborne.co.uk

Web www.epemag.com

For editorial contact details see page 7.

Next Month – in the February issue

Audio DSP – Part 2

In the second instalment, we will finish describing the circuit and present the parts list and board assembly instructions for our new, very versatile Hi-Fi stereo digital signal processor.

Motion-Triggered 12V Switch

This simple circuit switches on a 12V circuit when it detects acceleration or vibration.

A stamp-sized digital audio player

The DFPlayer Mini is a low-cost digital audio player module available from popular Internet suppliers. Despite its size and price, it can do a lot!

USB Keyboard and Mouse Adaptor for Micros

How can you connect a keyboard or mouse to a microcontroller, especially now that most keyboards and mice have a USB plug? This circuit is the answer. It's small, easy to build and it won't break the bank!

Colour Maximite Computer – Part 4

In the final part of this project we will cover graphical commands, demonstration programs and games, and how to use software to control externally connected hardware.

Using Stepper Motors

Explore the features of advanced bipolar stepper driver breakout boards.

PLUS!

All your favourite regular columns from *Audio Out*, *Cool Beans* and *Circuit Surgery*, to *Electronic Building Blocks*, *Techno Talk* and *Net Work*.



On sale 2 January 2020

Content may be subject to change

WELCOME TO JPG ELECTRONICS

Selling Electronics in Chesterfield for 29 Years

**Open Monday to Friday 9am to 5:30pm
And Saturday 9:30am to 5pm**

- Aerials, Satellite Dishes & LCD Brackets
- Audio Adaptors, Connectors & Leads
- BT, Broadband, Network & USB Leads
- Computer Memory, Hard Drives & Parts
- DJ Equipment, Lighting & Supplies
- Extensive Electronic Components
- ICs, Project Boxes, Relays & Resistors
- Raspberry Pi & Arduino Products
- Replacement Laptop Power Supplies
- Batteries, Fuses, Glue, Tools & Lots more...



T: 01246 211 202

E: sales@jpgelectronics.com

**JPG Electronics, Shaw's Row,
Old Road, Chesterfield, S40 2RB**

W: www.jpgelectronics.com



Retail & Trade Welcome • Free Parking • Google St View Tour: S40 2RB

NEW subscriptions hotline!

Practical Electronics

We have changed the way we sell and renew subscriptions. We now use 'Select Publisher Services' for all print subscriptions – to start a new subscription or renew an existing one you have three choices:

1. Call our NEW print subscription hotline:
01202 087631, or email: pesubs@selectps.com
2. Visit our online shop at: www.epemag.com
3. Send a cheque (payable to: 'Practical Electronics') with your details to:
Practical Electronics Subscriptions, PO Box 6337,
Bournemouth BH1 9EH, United Kingdom

Remember, we print the date of the last issue of your current subscription in a box on the address sheet that comes with your copy.

Digital subscribers, please continue to call 01202 880299 or visit: www.epemag.com

Published on approximately the first Thursday of each month by Electron Publishing Limited, 1 Buckingham Road, Brighton, East Sussex BN1 3RA. Printed in England by Acorn Web Offset Ltd., Normanton WF6 1TW. Distributed by Seymour, 86 Newman St., London W1T 3EX. Subscriptions UK: £26.99 (6 months); £49.85 (12 months); £94.99 (2 years). EUROPE: airmail service, £30.99 (6 months); £57.99 (12 months); £109.99 (2 years). REST OF THE WORLD: airmail service, £37.99 (6 months); £70.99 (12 months); £135.99 (2 years). Payments payable to 'Practical Electronics', Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom. Email: pesubs@selectps.com. PRACTICAL ELECTRONICS is sold subject to the following conditions, namely that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price shown on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter whatsoever.

Faster Programming, Wider Voltage Ranges and Enhanced Interface Options

PICkit™ 4 enables faster design development



With five times faster programming and a wider 1.2V to 5V range, the Microchip MPLAB® PICKit™ 4 Development Tool supports low-cost development of even more Microchip controllers.

PICKit™ 4 also introduces improved USB connectivity with a hi-speed USB 2.0 interface and advanced debugging over 4-wire JTAG and serial wire debug with streaming data gateway in addition to legacy interfaces.

The new advanced interfaces support in-circuit programming and debugging of Microchip's CEC1702 hardware cryptography-enabled MCUs, in addition to dsPIC® Digital Signal Controllers and PIC® microcontrollers from 8- to 32-bit.



www.microchip.com/PICKit4eu

DCA75

Advanced Semiconductor Analyser and Curve Tracer



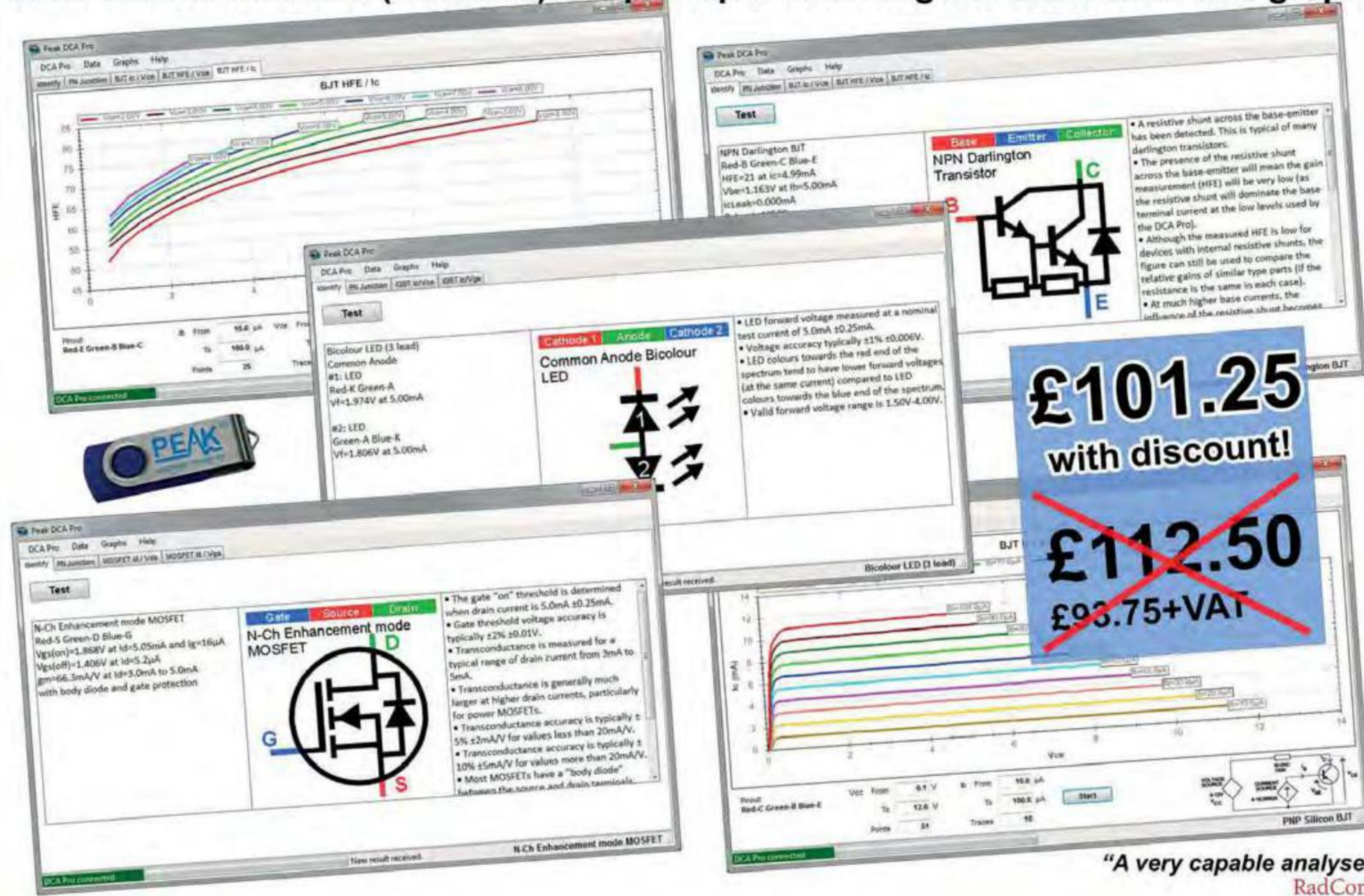
A versatile instrument that you'll love using for testing a huge range of semiconductor components.

Supports MOSFETs, transistors, diodes, LEDs, JFETs, IGBTs and more.

Connect any way round and let the **DCA Pro** identify the component type, pinout and lots of parameters.

Can be used standalone or connected to a Windows® PC (XP up to Win10).

Load the PC software (included) to open up a vast range of extra tests and graphs.



"A very capable analyser"
RadCom

It's only possible to show summary specifications here. Please ask if you'd like detailed data. Further information is also available on our website. Product price refunded if you're not happy.

Tel. 01298 70012
www.peakelec.co.uk
sales@peakelec.co.uk

Atlas House, 2 Kiln Lane
Harpur Hill Business Park
Buxton, Derbyshire
SK17 9JL, UK

